Master in Artificial Intelligence (UPC-URV-UB)

# Master of Science Thesis

# Using fuzzy methods for rule extraction in the discrimination of class C GPCR subtypes from their subsequences

Stavros Koulas

**Advisors: Dr. Àngela Nebot, Dr. Alfredo Vellido, Dr. Francisco Mugica**

September 5, 2014

## Acknowledgements

First and foremost I would like to express my gratitude to my advisors Dr. Àngela Nebot and Dr. Alfredo Vellido, as well as Dr. Francisco Mugica, members of the Soft Computing Research Group at the Department of Computer Science, for their introduction to the whole topic of research and also for their useful comments, feedback and in general all their engagement throughout the learning process of this master thesis.

Secondly, I would like to thank fellow student Christiana Halka for her help during the master thesis as well as Mrs. Martha Ivón Cárdenas, whose academic research both in the MSc as well as her ongoing PhD in the subject, has helped me understand the whole field in a better way.

I would also like to thank all the professors and faculty of the Master's degree on Artificial Intelligence, helping me all little by little to reach this point.

Last but not least, I thank my friends and family for their support during the master thesis and especially my parents for their financial support throughout my studies and for their endless encouragement.

## Further acknowledgements

This master thesis has been supported by and is part of a broader research effort substantiated as project TIN2012-31377: “KAPPA AIM: Knowledge Acquisition in Pharmacoproteomics using Advanced Artificial Intelligence Methods”, led by Dr. Alfredo Vellido and funded by the Spanish Ministry of Economy and Competitivity.

## Abstract

G-Protein-Coupled receptors (GPCR) are cell membrane proteins that regulate many of the cell functions and transduce signals between the intracellular and extracellular domains. This makes them relevant in pharmacology as therapeutic targets. As members of this superfamily, class C GPCRs in particular regulate a number of important physiological functions. Proteins of the class must be studied from their primary sequences, as only one of their 3-D structures has been fully determined, earlier this year. Protein function investigation requires the identification of motifs, or functional subsequences. In this thesis, we will describe the discrimination of class C GPCR subtypes through interpretable rules from a specific alignment free transformation of the sequences, namely amino acid composition. The Fuzzy Inductive Reasoning methodology was used as the basis to extract these linguistic rules.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

G-Protein Coupled Receptors (GPCRs) are cell membrane proteins of great relevance to biology in general and particularly to biology at the molecular and cellular levels, due to their role in transducing extracellular signals.

To provide readers with some context for the importance of research in this area, bear in mind that the 2012 Nobel Prize in Chemistry was awarded to researchers Brian Kobilka and Robert Lefkowitz for their work which was "crucial for understanding how G-protein–coupled receptors function" [10].

GPCRs share a common architecture that has been overall conserved over the course of evolution despite natural variation. Note that many present-day eukaryotes, including animals, plants, fungi, and protozoa, rely on these receptors to receive information from their environment.

For example, simple eukaryotes such as yeast have GPCRs that sense glucose and mating factors. Not surprisingly, GPCRs are involved in considerably more functions in multicellular organisms. Humans alone have nearly 1,000 different GPCRs, and each one is highly specific to a particular signal [9].

Recent research has shown an incremental interest in GPCRs as they are of extreme paramount in the quest for new medicines, and given that new functions for GPCRs are continuously discovered -especially for the orphan GPCRs for which no function is currently known- the number of drugs that target GPCRs can only be expected to increase.

Proteins of the class C family of GPCR must still be studied mainly from their primary residue sequences, as none bar one of their 3-D structures has yet been described in full. Protein function investigation requires -at different levels- the identification of motifs, or functional subsequences from the primary protein information.

The focus of this thesis is on the Class C GPCR subtype, which includes a number of different cell surface receptors such as, amongst others, metabotropic glutamate receptors, extracellular calcium-sensing receptors, gamma-amino-butyric acid (GABA) type B receptors and vomeronasal type-2 receptors.

The main objective of this thesis is the study of the discrimination between class C GPCR subtypes through interpretable rules based on their amino acid (AA) sequence.

The importance of interpretability should not be underestimated: despite the obvious relevance of classical figures of merit (accuracy in its many variants, ROC plots and the like), an accurate discrimination that cannot be described in ways that are intuitively understandable by a human expert in the application area (bioinformatics and pharmaco-proteomics in our case) is unlikely to evolve into actionable research results [47].

Automatically-extracted rules describing given data subgroups or classes have previously been shown to be an effective way to reach this interpretability target in biomedical applications [48] [49].

The remaining of this thesis is structured as follows:

- Chapter 2 provides the reader with a self-contained summary description of G-protein-coupled receptors and their importance in current pharmaco-proteomics research.
- Chapter 3 introduces the research materials on which the thesis experimentation is based as well as some detailed information regarding the data set. Furthermore, the reader is also informed regarding related work in the area of GPCRs, as well as the application of rule extraction methods in biomedical problems.
- Chapter 4 gears into Fuzzy Inductive Reasoning, including theoretical background of the methodology as well as detailed information about the algorithm steps and configuration options of the VisualFIR software, used for the extraction of results.
- Chapter 5 accounts for the experimental work regarding rule extraction, as well as an experimental classification approach, and the discussion of the results obtained.

- Chapter 6, finally, concludes on the results of the thesis and proposes feasible future work in using fuzzy methods regarding the GPCR area and the research materials used.

# Chapter 2

# G-Protein Coupled Receptors

## 2.1 The biological background

G-Protein Coupled Receptors (GPCRs) are cell membrane proteins of great relevance to biology systems and proteomics due to their role in transducing many extracellular signals to the intracellular domain.

Most physiological processes depend on GPCRs and around half of all medical drugs act through these receptors.

The presence of GPCRs in the genomes of bacteria, yeast, plants, nematodes and other invertebrate groups argues in favor of a relatively early evolutionary origin of this group of molecules.

The diversity of GPCRs is dictated both by the multiplicity of stimuli to which they respond, as well as by the variety of intracellular signaling pathways they activate. These include light, neurotransmitters, odorants, biogenic amines, lipids, proteins, amino acids, hormones, nucleotides, chemokines and, undoubtedly others yet to be discovered.

## 2.2 GPCRs as pharmacological targets

Recent research has shown that metabotropic glutamate (mGlu), gamma-aminobutyric acid (GABA) of type B ($GABA_B$) and calcium sensing (CaS) receptors represent an important new class of therapeutic targets that are integral to disorders that affect the central neural system (CNS) and calcium homeostasis [26].

Importantly, more than a third of all drugs approved by the US Food and Drug Administration over the last three decades actually target GPCRs [4], which makes them an obvious and highly desirable target -of large scale- research in the pharmaceutical industry.

A decade ago, only 10% of GPCRs were known drug targets according to Vassiliatis et al. [23]. But as new functions for GPCRs are discovered, especially for the orphan GPCRs for which no function is currently known, the number of drugs that target GPCRs can only be expected to increase. This is a focus of intense research effort, both in academia and in industry.

According to the most recent research, the percentage has dramatically increased up to even 40% in 2013 [22].

These cell surface receptors act like an inbox for messages in the form of light energy, peptides, lipids, sugars and proteins. Such messages inform cells about the presence or absence of life-sustaining light or nutrients in their environment, or they convey information sent by other cells [9].

This is extremely important for protein homology detection. Given that, despite research efforts, the complete 3D structure and even the functionality of most GPCRs has yet to be clarified, the construct of robust classification models for analysis based on their AA sequence [1] becomes an urgent matter for research. Importantly, several databases of GPCR primary sequences are publicly available [5].

In addition to biological studies of the types summarized above, much excitement remains in the field because of the continuing de-orphanization of GPCRs and the subsequent elucidation of their pharmacology and physiology.

Once a large enough panel of GPCRs has been obtained and comprehensively characterized, a systematic analysis of the `*receptorome*' (the portion of the proteome encoding receptors) can yield important discoveries.

Such approaches have been used to discover the molecular mechanisms responsible for serious drug side-effects – for example, phen/fen-induced heart disease and weight gain associated with the use of atypical antipsychotics [24][25]. Additionally, the screening of the *receptorome* has been used to elucidate the actions of natural compounds and to obtain validated molecular targets for drug discovery [6].

It has also been made evident that GPCRs are responsible for regulating hormone secretions in the pancreas [27].

So far, two drugs acting at family C receptors (the $GABA_B$ agonist *Baclofen* and the positive allosteric CaR modulator (*Cinacalcet*) have been marketed.

*Cinacalcet* is the first allosteric GPCR modulator to enter the market, which demonstrates that the therapeutic principle of allosteric modulation can also be extended to this important drug target class [15].

$GABA_B$ receptors are known to control neuronal excitability and modulate synaptic neurotransmission, playing a very important role in many physiological activities. These receptors are widely expressed and distributed in the nervous system and have been implicated in a variety of neurodegenerative and pathophysiological disorders including epilepsy, spasticity, chronic pain, depression, schizophrenia and drug addiction [16].

## 2.3 Receptors

Cell surface receptors (membrane receptors, transmembrane receptors) are specialized integral membrane proteins that take part in communication between the cell and the outside world. Extracellular signaling molecules (for instance, hormones, neurotransmitters, cytokines, growth factors or cell recognition molecules) attach to the receptor, triggering changes in the cell function. This process is called signal transduction: The binding initiates a chemical change on the intracellular side of the membrane.

In this way, receptors play a unique and important role in cellular communications and signal transduction.

Like any integral membrane protein, a transmembrane receptor may be subdivided into three parts or domains.

- **Extracellular domain**

The extracellular domain is the part of the receptor that sticks out of the membrane on the outside of the cell or organelle. If the polypeptide chain of the receptor crosses the membrane bilayer several times, the external domain can comprise several "loops" sticking out of the membrane.

By definition, a receptor's main function is to recognize and respond to a specific ligand, for example, a neurotransmitter or hormone (although certain receptors respond also to changes in transmembrane potential), and in many receptors these ligands bind to the extracellular domain.

- **Transmembrane domain**

In the majority of receptors for which structural evidence exists, transmembrane *alpha* helices make up most of the transmembrane domain. In certain receptors, such as the nicotinic acetylcholine receptor, the transmembrane domain forms a protein-lined pore through the membrane, or ion channel.

Upon activation of an extracellular domain by binding of the appropriate ligand, the pore becomes accessible to ions, which then pass through. In other receptors, the transmembrane domains are presumed to undergo a conformational change upon binding, which exerts an intracellular effect.

In some receptors, such as members of the 7 transmembrane (TM) superfamily to which GPCRs belong, the TM domain may contain the ligand binding pocket; evidence for this has been determined by crystallography.

- **Intracellular domain**

The intracellular (or cytoplasmic) domain of the receptor interacts with the interior of the cell or organelle, relaying the signal. There are two fundamentally different ways for this interaction:

The intracellular domain communicates via specific protein-protein-interactions with effector proteins, which in turn send the signal along a signal chain to its destination.

With enzyme-linked receptors, the intracellular domain has enzymatic activity that can be located on an enzyme associated with the intracellular domain.

**Signal Transduction**

Signal transduction processes through membrane receptors involve external reactions, in which the ligand binds to a membrane receptor, and internal reactions, in which intracellular response is triggered [19].

Based on structural and functional similarities, membrane receptors are mainly divided into three classes: The ion channel-linked receptor, the enzyme-linked receptor and GPCRs.

In Figure 2.1, we can see the illustration of a receptor at the very moment when it transfers the signal from a hormone on the outside of the cell to the G-protein on the inside of the cell.

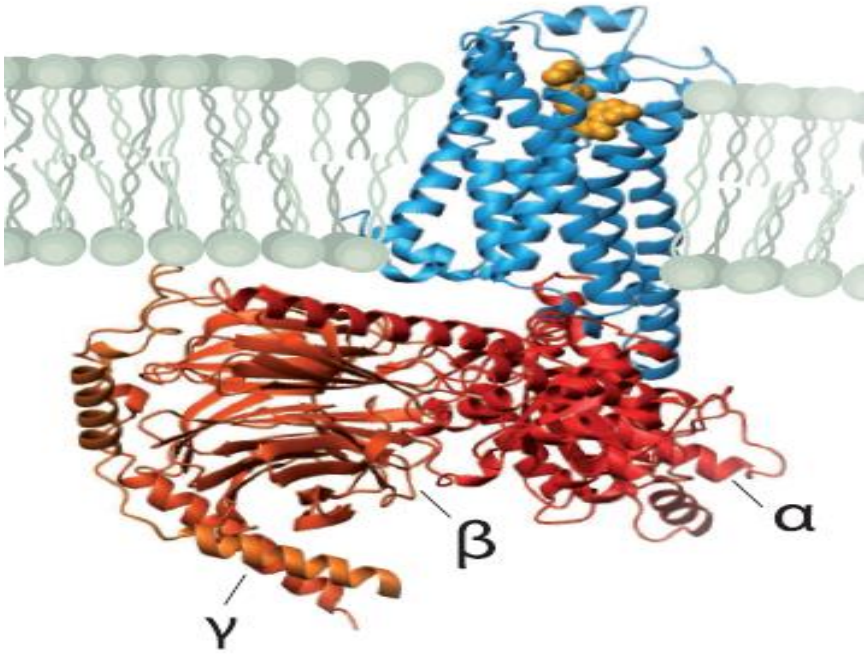


Figure 2.1. Illustration of Kobilka's crystal structure of an activated β-adrenergic receptor (blue). A hormone (in orange) attaches to the outside to the GPCR (in blue) and a G-protein (in red) couples on the inside.

Most ligands responsible for cell-cell signaling bind to receptors on the surface of their target cells. Consequently, a major challenge in understanding cell-cell signaling is unraveling the mechanisms by which cell surface receptors transmit the signals initiated by ligand binding. Some neurotransmitter receptors are ligand-gated ion channels that directly control ion flux across the plasma membrane plasma membrane. Other cell surface receptors, including the receptors for peptide hormones and growth factors, act instead by regulating the activity of intracellular proteins. These proteins then transmit signals from the receptor to a series of additional intracellular targets. Ligand binding to a receptor on the surface of the cell thus initiates a chain of intracellular reactions, ultimately reaching the target cell nucleus and resulting in programmed changes in gene expression [28].

**General Mechanism of Signal Transduction through GPCR and G Proteins**

The regulatory cycle of G proteins i.e., activation/inactivation through GPCR is shown in Figure 2.1. In the inactive state, $G_{\alpha}$ is bound to $G_{\beta\gamma}$ dimer and GDP. G protein mediated signaling starts by binding of an agonist molecule that leads to activation of GPCR.

GPCR is also a guanine nucleotide exchange factor that promotes the exchange of guanosine disphosphate (GDP)/guanosine triphosphate (GTP) associated with the $G_{\alpha}$ subunit [21]. Therefore, the activated GPCR catalyzes exchange of GTP for GDP on the $G_{\alpha}$ subunit, as a result conformational changes takes place in the GPCR, which leads to dissociation of $G_{\beta\gamma}$ dimer from $G_{\alpha}$ and thus activates multiple molecules of G proteins (Figure 2.1).

The G proteins activated in this way constitute an amplified representation of the activated GPCR. Activated $G_{\alpha}$ and $G_{\beta\gamma}$ proteins in turn binds to various effectors and thereby switches it either on or off in different systems, and effectors continue to pass the signal to different kinds of second messengers.

Here intrinsic GTPase activity of $G_{\alpha}$ comes into play, that leads to conversion of bound GTP into GDP and hence the inactivation of G proteins cascade. GTPase activity of the $G_{\alpha}$ subunits may also be regulated by regulators of G proteins signaling (RGS proteins) as well

as effectors. Moreover, effector enzymes such as adenylyl cyclases may also regulate the activation of G proteins by receptors [20].

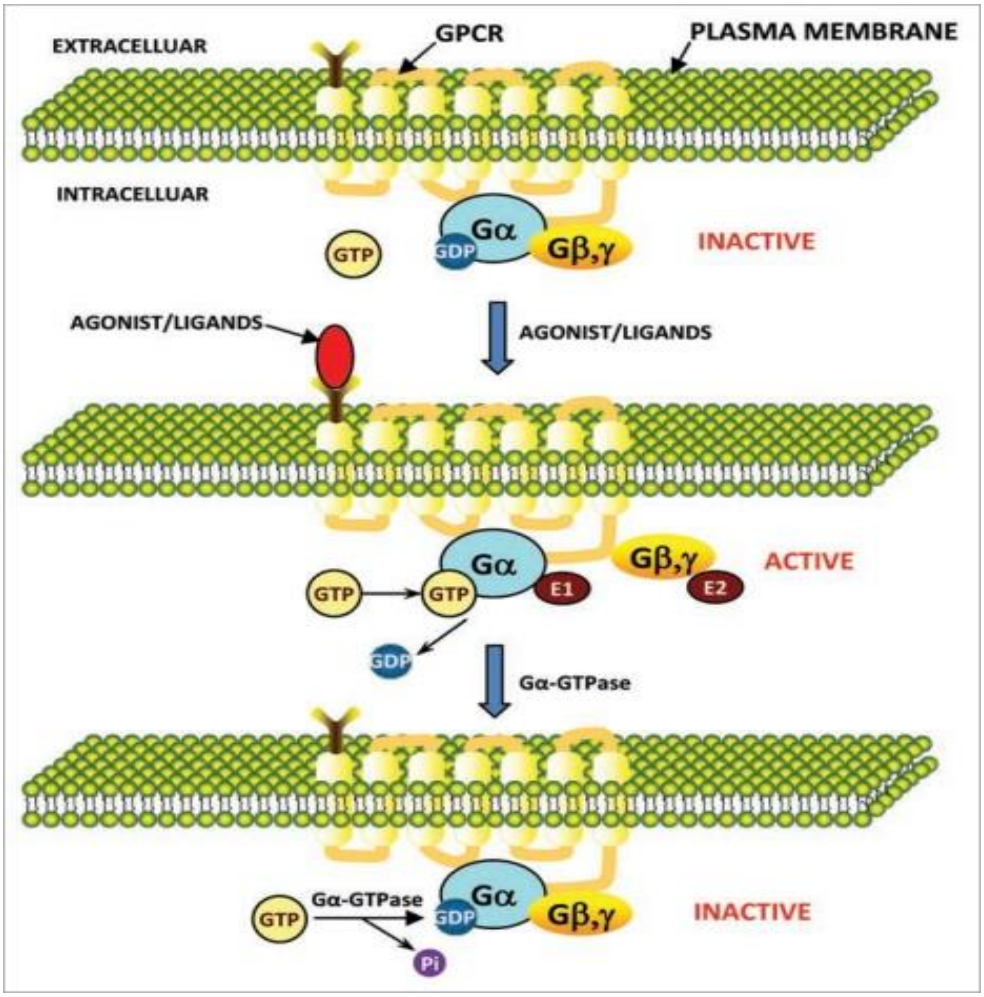


Figure 2.2. Model for signal transduction by activation/inactivation of heterotrimeric G proteins through GPCR [20]

On the basis of homology with rhodopsin, they are predicted to contain seven membrane-spanning helices, an extracellular N-terminus and an intracellular C-terminus. This gives rise to their other names, the 7-TM receptors or the heptahelical receptors - explaining why GPCRs are sometimes called seven-transmembrane receptors; and the intervening portions loop both inside and outside the cell. The extracellular loops form part of the pockets at which signaling molecules bind to the GPCR.

### 2.3.1 GPCRs: Structure, function and classification

As stated, GPCRs primary function is to transduce extracellular stimuli into intracellular signals and as part of the broader G protein family, GPCRs have the ability to bind the nucleotides GTP and GDP.

Some G proteins, such as the signaling protein $R_{as}$, are small proteins with a single subunit. However as discussed above, GPCRs have different subunits and two of the three different GPCR subunits (since the G proteins that associate with GPCRs are heterotrimeric, meaning they have three different subunits) -alpha and gamma- are attached to the plasma membrane by lipid anchors as seen in Figure 2.3.

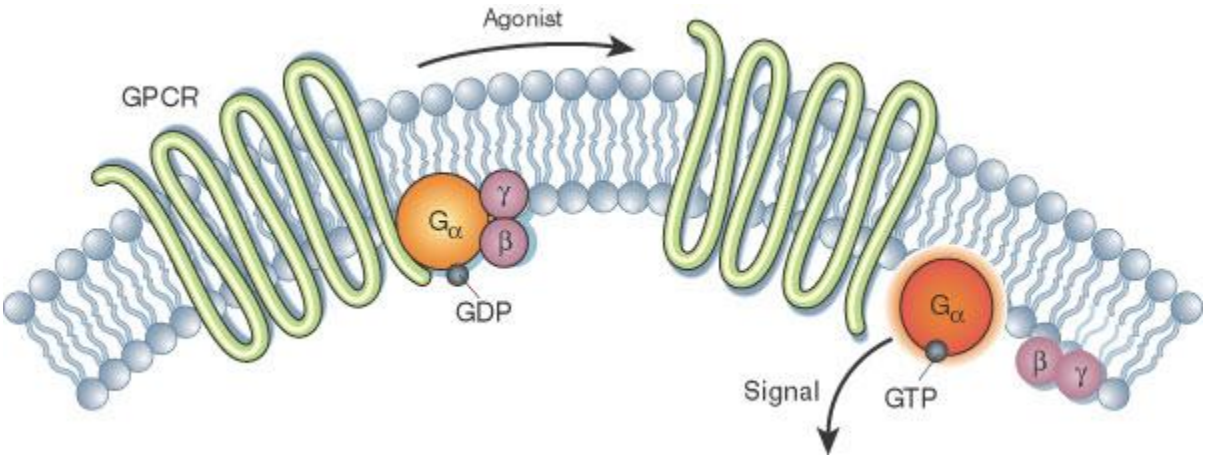


Figure 2.3: Activation of the G alpha subunit of a G-protein-coupled receptor [9]

In unstimulated cells, the state of G alpha (orange circles) is defined by its interaction with GDP, G beta-gamma (purple circles), and a G-protein-coupled receptor (GPCR; light green loops). Upon receptor stimulation by a ligand called an agonist, the state of the receptor changes. G alpha dissociates from the receptor and G beta-gamma, and GTP is exchanged for the bound GDP, which leads to G alpha activation. G alpha then goes on to activate other molecules in the cell.

#### 7-TM Receptors

As seen in Figure 2.2 and Figure 2.3, these 7-TM receptors are integral membrane proteins that contain seven membrane-spanning helices. As the name suggests they are coupled to

heterotrimeric G proteins on the intracellular side of the membrane. Upon ligand binding, the GPCR undergoes a conformational change which is transmitted to the G protein causing activation. Further signal transduction depends on the type of G protein.

However, recently it has become apparent that other receptors and proteins that are not heptahelical or serpentine also mediate some of their biological effects via activation of heterotrimeric G proteins. To note, the role of heterotrimeric G proteins in mediating the actions of these non-classical GPCRs such as receptors for a variety of growth factors, atrial natriuretic hormone, extracellular matrix proteins, as well as zona pellucida glycoprotein ZP3 has not been elucidated [13].

The human genome encodes roughly 350 7-TM receptors, of which approximately 150 of these have an unknown function. They comprise the largest family of receptors in the human genome and because of their large number, widespread distribution and important roles in cell physiology and biochemistry; they play an important role in medicine.

Diseases involving mutations of 7-TM receptors are relatively rare. However disorders which involve antibodies directed against 7-TM receptors are more common. Typically disease can be caused by:

- Non-functional receptors (GHRH and familial growth hormone deficiency)
- Constitutive activation (Rhodopsin and retinitis pigmentosa)
- Changes in ligand binding specificity (Thyroid-stimulating hormone receptor and hyperthyroidism of pregnancy)
- Improper receptor processing (vasopressin receptor and diabetes insipidus)
- Antibodies directed against the receptors (Thyroid stimulation hormone receptor and Graves' disease)
- Constitutively active or inactive G proteins

7-TM receptors are the target of around half of all modern medicinal drugs. Their expression on the cell surface makes them readily accessible to hydrophilic drugs and their non-uniform expression provides selectivity in activating or blocking physiological events.

Agonists and antagonists of 7-TM receptors are used in the treatment of disease in every organ system [8].

**The GPCRs family**

GPCRs are the largest class of receptors, with more than one thousand GPCRs identified so far. The GPCRs family is the third most abundant family in Caenorhabditis elegans, comprising 5% of its genome with approximately 1,100 members. The Drosophila genome has at least 160 GPCRs. Based on the now entirely known human genome, careful estimation suggest that about 3–4% of the human genes code for GPCRs, about 1,200–1,300 members of GPCR superfamily in the human genome, many of which are known to homo- and heterodimerize. However, in case of plants a single GPCR has been isolated from pea27 and maize (Acc. No. NM_001153424), and computational analysis show their presence in Arabidopsis, Populus and rice [20].

A popular database of GPCRs, the GPCRDB [2], divides the GPCR subfamily into five major classes alphabetically -from A to E- based on the ligand types, functions and sequence similarities.

Some other approaches, like Papasaikas et al. [37] suggested that GPCR recognition and classification at the family level can also be analyzed using probabilistic methods such as Hidden Markov Models, in order to determine to which GPCR family a query sequence belongs or resembles.

Inevitably, estimates of the number of GPCRs in the human genome vary widely based on their sequences, as well as on their known or suspected functions. There are five or even six major classes of GPCR based on sequence homology and functional similarity among each other.

- **Family A (rhodopsin receptor family)**

Family A is commonly known as *rhodopsin family*. It is the largest (and possibly the best known) family of GPCRs and includes receptors for odorants and small ligands. This family is further divided into three groups. Group 1 contains GPCRs for small ligands including rhodopsin and β-adrenergic receptors. The binding site is localized within the seven TMs. Group 2 contains receptors for peptides whose binding site includes the N-terminal, the extracellular loops and the superior parts of TMs. Group 3 contains GPCRs for glycoprotein hormones.

- **Family B (secretin receptor family)**

Family B is commonly known as *secretin family*. It includes about 60 members and is characterized not only by the lack of the structural signature present in family A but also by the presence of a large N-terminal ectodomain. Family B GPCRs have a similar morphology to group A3 GPCRs, but they do not share any sequence homology. Their ligands include high molecular weight hormones such as glucagon, secretine, calcitonin, growth hormone-releasing hormone, corticotropin-releasing factor, VIP-PACAP and the Black widow spider toxin, α-latrotoxin.

- **Family C (metabotropic glutamate/pheromone receptors family)**

Family C consists of about two dozen GPCRs such as mGluR and the Ca receptors. This family also includes $GABA_B$ receptors, taste receptors, olfactory receptors and a group of putative pheromone receptors coupled to the G protein $G_o$ (termed VRs and $G_o$-VN). Like family B, these receptors possess large ectodomains responsible for ligand binding.

- **Family D (fungus pheromone receptor family)**

Family D comprises pheromone receptors (VNs) associated with $G_i$.

- **Family E (Cyclic AMP receptor family)**

Cyclic AMP (OR cyclic-cAR) receptors have only been found in D.discoideum, but its possible expression in vertebrate has not yet been reported.

- **Family F (frizzled/smoothened receptor family)**

Family F includes the *frizzled* and the *smoothened* (Smo) receptors involved in embryonic development and in particular in cell polarity and segmentation.

These GPCR families are further classified in up to 64 different subfamilies based on the multiple receptor subtypes, each of which is encoded by separate genes. Each subfamily is further subdivided into different groups, based mainly on the TiPS classification scheme that takes into account the native ligand(s) that binds to a particular GPCR. Therefore, GPCRs exist as a big family of receptors that binds to a vast variety of ligands and are involved in various signaling pathways.

### 2.3.2 GPCR Family C

As mentioned, GPCRs have become an important research target for new therapies for pain, anxiety, neurodegenerative disorders and as antispasmodics [1] and the recently identified class (or family) C of GPCRs has been targeted by two therapeutic drugs currently on the market [16].

The aim of this thesis was to focus on a subtype of the GPCR family, class C whose 3D structure is unknown to date.

As discussed in the previous section, whereas all GPCRs are characterized by sharing a common seven TM helices domain (7TM), responsible for G protein activation, most class C GPCRs include in addition a large extracellular domain (ECD), the Venus Flytrap (VFT) and a cysteine rich domain (CRD) connecting both [3].

Class C GPCRs play important roles in many physiological processes such as synaptic transmission, taste sensation and calcium homeostasis, and include mGlu, $GABA_B$, CaS, taste 1 receptors (TAS1), as well as a few orphan receptors. A distinguishing feature of class C GPCRs is their constitutive homo- or hetero-dimerization mediated by a large N-terminal ECD (See Figure 2.4).

The ECD of class C GPCRs consists of a VFD which contains the orthosteric binding site for native ligands (Figure 2.4) and a CRD with the exception of $GABA_B$ receptors. The CRD, which mediates the communication between ECD and 7-TM domains, is stabilized by disulfide bridges, one of which connects the CRD and VFD.

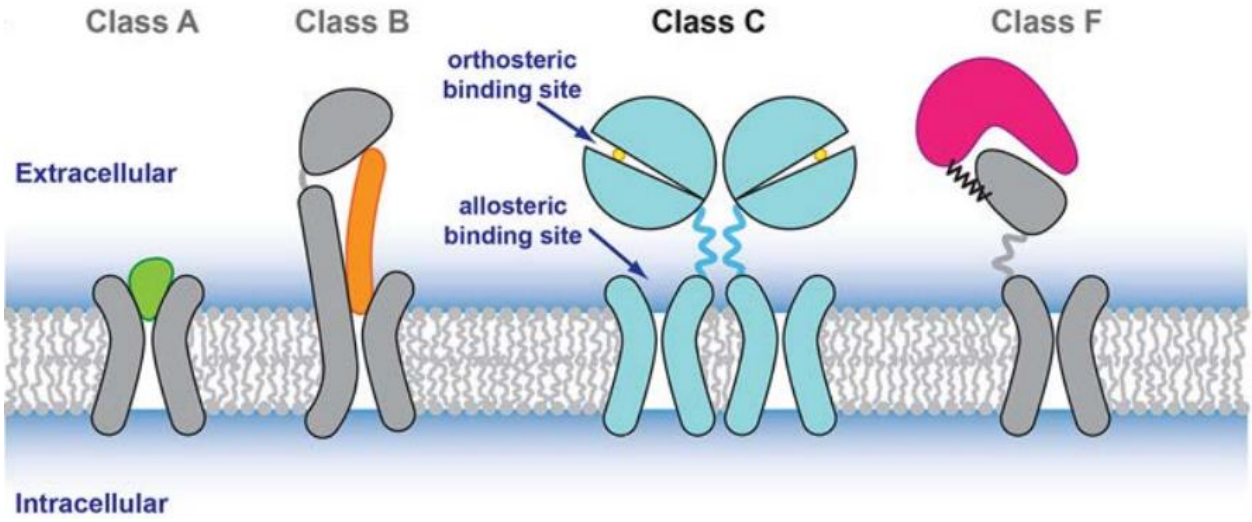


Fig. 2.4. Overall structure of the *mGlu1* TM domain [41].

For class A, in most cases, the endogenous ligand (shown in green) is recognized by an orthosteric site in the 7TM domain. For class B, the endogenous peptide ligand (shown in orange) binds to both ECD and 7TM domains. For class C, the endogenous small molecule ligands (shown as yellow circles) are recognized by orthosteric sites in the VFDs. For class F, lipoprotein WNT (shown in magenta) binds the CRD domain of Frizzled receptors.

### 2.3.3 Metabotropic glutamate receptors

The mGlu family was the first group of class C GPCRs to be cloned in 1991 [46]. Comprised of eight members, the mGlu family can be separated into three subgroups, termed *groups I* (mGlu1 and mGlu5), *II* (mGlu2 and mGlu3) and *III* (mGlu4,6,7,8), based on their sequence homology, G protein coupling profile, and signal transduction pathways and pharmacology [45].

The mGlu *group I*, mGlu1 and mGlu5, are considered promising therapeutic targets to treat diseases including cancer, pain, schizophrenia, Alzheimer's disease, anxiety, and autism. However, the development of subtype-selective small molecule ligands that might serve as drug candidates for these receptors has been hampered by the conservation of the orthosteric (glutamate) binding site. This problem can be over-come using allosteric modulators that act at alternative binding sites; these compounds bind predominantly within the 7TM domain of the class C receptors. Allosteric modulators can alter the affinity or efficacy of native ligands in positive, negative, and neutral ways, demonstrating a spectrum of activity that cannot be achieved by orthosteric ligands alone.

The metabotropic glutamate receptors are functionally and pharmacologically distinct from the ionotropic glutamate receptors. They are coupled to G-proteins and stimulate the inositol phosphate/$Ca^{2+}$ intracellular signalling pathway.

The orthosteric sites of mGlu receptor subtypes are the most highly conserved throughout evolution, such that there are almost no orthosteric ligands that display higher selectivity for a given subtype. Moreover, the glutamate-binding pocket strictly selects for agonists with amino acid-like structures, which are notoriously difficult to synthesize and display undesirable pharmacokinetics. By contrast, most of the allosteric modulators for mGlu receptors possess better subtype selectivity as a result of less conserved allosteric sites and better pharmacological properties due to their structural diversity and more extensive lipophilic nature [18].

The first allosteric modulator that was discovered for class C GPCRs is CPCCOEt, which functions as a numerous allosteric modulator (NAM) for the mGlu1 receptor. Numerous allosteric modulators of *group I* mGlu receptors have since been identified.

The allosteric modulators of the mGlu5 receptor are leading with regard to the development of pharmaceuticals that target class C GPCRs. Convincing preclinical data have shown a significant effect of several positive allosteric modulators (PAMs) in schizophrenia and furthermore, positive clinical results have also been obtained for NAMs in *L*-DOPA-induced tardive dyskinesia in Parkinson's disease [19][38].

Most allosteric modulators for *group II* mGlu receptors are PAMs. PAMs with selectivity for the *mGlu2* receptor have displayed similar effects as agonists in an animal mode [39], which suggests that there is a high possibility for success in clinical trials.

Compared with the modulators that have been described for *groups I* and *II* mGlu receptors, notably fewer allosteric modulators have been identified that target *group III*.

It is also important to note that some allosteric modulators that target *group I* have the opposite effect on *group III*. Recently, the *mGlu4* receptor has been the focus of significant attention because the corresponding PAMs that target this receptor represent promising novel drugs with which to treat Parkinson's disease [38].

# Chapter 3

# Research Materials

### 3.1 The GPCR Dataset

The data set investigated in this thesis was obtained from GPCRDB [43] as of March 2011. This is a curated database that aims to gather and manage heterogeneous data on GPCRs and contains data not only data on sequences, but also information on ligand binding constants, mutations and computationally derived data (such as multiple sequence alignments and homology models).

Much current research on GPCRs concerns the use of multiple-aligned sequences. Despite is advantages, sequence alignment implies that we must renounce to the use of a sizeable amount of sequential information. Alternatively, we can consider analyzing these data from the unaligned full sequences. Computational intelligence (CI) techniques, though, are not usually suitable for raw alignment-free sequences and they must be transformed in one way or another. The transformation of the symbolic sequences into real-valued feature vectors can for instance be accomplished on the basis of the physicochemical properties of their constituent amino acids.

In this thesis, though, we focus on a very simple transformation method that only accounts for the frequency of appearance of each residue (amino acid type) in the sequence. Despite the fact that such representation does not make explicit use of the sequential information, it has previously been shown to yield reasonably good results in GPCR subtype discrimination tasks, both in semi-supervised and supervised settings [42] [50].

The strategy of focusing in such simple transformation is clear and has already been sketched in the introductory chapter: Despite the fact that the main objective of this thesis is the study of the discrimination between class C GPCR subtypes through interpretable rules based on their AA sequence, we are specifically interested in achieving rule interpretability so that human experts in bioinformatics and pharmaco-proteomics can use them as

actionable research results. Interpretable rules require a reasonably parsimonious set of observed variables to start with, and starting from a single variable for each residue in the AA alphabet seems a reasonable compromise.

The data used for the experiments are thus encoded as 20 input variables that represent the AA sequence of the protein, together with a target variable that indicates which of the seven GPCR subtypes the sequence belongs to.

As previously mentioned, the AA sequence transformation considers only the relative frequencies of appearance of the 20 AAs, shown in Table 3.1, thus ignoring the sequential order. The original lengths of the analyzed sequences vary from 250 to 1995 AAs (which is an indication of the interest in using an alignment-free strategy).

| Single-Letter code | Abbreviation | Full Name |
|---|---|---|
| A | Ala | Alanine |
| R | Arg | Arginine |
| N | Asn | Asparagine |
| D | Asp | Aspartic acid |
| C | Cys | Cysteine |
| Q | Gln | Glutamine |
| E | Glu | Glutamic acid |
| G | Gly | Glycine |
| H | His | Histidine |
| I | Ile | Isoleucine |
| L | Leu | Leucine |
| K | Lys | Lysine |
| M | Met | Methionine |
| F | Phe | Phenylalanine |
| P | Pro | Proline |
| S | Ser | Serine |
| T | Thr | Threonine |
| W | Trp | Tryptophan |
| Y | Tyr | Tyrosine |
| V | Val | Valine |

Table 3.1. Alphabet of the twenty amino acids in the transformed dataset.

The data used for class C GPCRs, consists of 1,510 sequences that are further subdivided into 7 categories/subtypes. These subtypes and the number of cases belonging to each of them are listed in Table 3.2.

| | |
|---|---|
| Metabotropic Glutamate (MGlu) | 351 |
| Calcium sensing | 48 |
| GABA-B | 208 |
| Vomeronasal | 344 |
| Pheromone | 392 |
| Odorant | 102 |
| Taste | 65 |

Table 3.2: The seven GPCR subtypes and their corresponding number of sequences in the GPCRDB database.

### 3.2 The mGluR Dataset

In our experiments, we are also interested in the mGluR subtype in particular. It is the second biggest subtype of Class C GPCR in the database, including 351 instances. By removing the missing values, we end up with a subset containing 321 instances.

The mGluR subtype, of special interest in pharmacology, is further subdivided into eight subtypes, together with a group of mGluR-like sequences. These are listed in Table 3.3, together with the corresponding number of sequences for each subtype.

| | |
|---|---|
| mGluR1 | 32 |
| mGluR2 | 25 |
| mGluR3 | 34 |
| mGluR4 | 19 |
| mGluR5 | 29 |
| mGluR6 | 15 |
| mGluR7 | 4 |
| mGluR8 | 98 |
| mGluR-like | 65 |

Table 3.3: The eight mGluR subtypes and their corresponding number of sequences in the GPCRDB database, together with mGluR-like subset of sequences.

Discarding the mGluR-like subtype, mGluR can be grouped into three main categories [45]:

**Group1**: mGluR1 & mGluR5;

**Group2**: mGluR2 & mGluR3;

**Group3**: mGluR4 & mGluR6 & mGluR7 & mGluR8.

### 3.3 Related Work

Recent analysis of Class C GPCR sequence data using semi-supervised classification and the AA transformation that we have also used in this thesis, showed that accuracy reaches an upper bound at between 80-85% [41] that is not significantly increased when more sophisticated physicochemical transformations of the sequence are used (with results under the level of 90% accuracy) [5].

Although one might question that the simplicity of the dataset might risk losing relevant information, studies using Support Vector Machines (SVM) have shown a promising -and best in the area- classification of 88% [42]. Latest research results show accuracies in the area of 93-94% using the *digram* (a particular case of the more general *n-gram*) transformation (which considers the frequencies of occurrence of any given pair of AAs) [44].

Regarding the mGluR dataset, recent research using Generative Topographic Maps (GTM) [44] has shown that while most subtypes show a reasonable level of separation, none of them avoids some level of subtype overlapping. More specifically, mGlu8 have been found to be separated in 4 groups, mGlu in two groups and mGlu2 as single concentrated group. However, it is evident that different visual representations can lead to various results, as well as to different class-entropy levels for different data transformations [44].

# Chapter 4

# Research Methodology

### 4.1 Fuzzy Inductive Reasoning

Fuzzy Inductive Reasoning (FIR), derived from the General Systems Problem Solver (GSPS) [32], is a methodological tool for data-driven construction of dynamical systems and for studying their conceptual modes of behavior. FIR performs induction starting from raw data to build qualitative models. Some of the main advantages of this methodology are the following:

- The methodology can be applied to any domain. It is fully pattern–based, with no need for assuming any internal structure of the constructed system. In this respect, it is similar to artificial neural networks.

- FIR allows the otherwise qualitative models to treat time as a continuous (quantitative) variable. This is of primary importance when dealing with mixed quantitative/qualitative systems. In this respect, other qualitative approaches, such as the Qualitative Physics of Forbus [12] or the QSIM of Kuipers [57], are not in this case adequate.
- The methodology contains an inherent model validation mechanism that prevents reaching conclusions that are not justifiable on the basis of available facts. In this respect, FIR is similar to knowledge–based systems.
- Inductive reasoning operates in a qualitative fashion just like the knowledge–based reasoning. Although it is not able to offer a complete trace back of the full reasoning process, as expert systems do, it does provide information about the subset of variables selected for the reasoning process, and it can at least provide a justification for the predicted output based on the qualitative states of the selected input variables. Somehow, the structure of the system that allows us to provide explanations underlay in the set of pattern rules generated by inductive reasoning. There are various ways of generalizing this structure, for example using linguistic or fuzzy rules, as explained in Section 4.3.

In general, allowing more uncertainty tends to reduce complexity and increase credibility of the resulting model. In fact, this principle has guided the development of FIR, with the aim of enlarging the class of problems that can be dealt with by FIR [58].

**General Systems Theory (GST)**

The expression of similarity in the form of algebraic or differential equations is a kind of mathematical isomorphism. When this is generalized to include any relation, whether expressible by equations or not, then the concept of general systems acquires its proper meaning [12].

L. Von Bertalanffy, a known biologist, conceived the idea of GST when he generalized, within the framework of his theory of open systems, a number of principles for the construction of modern theoretical biology. This idea -theory of open systems- was developed according to the conclusion that the classical concept of a close system- commonly used in physics- was not useful and often leads to incorrect conclusions in the biological field. From this point of view, the organism is a certain "system" possessing organization and wholeness and is constantly changing, maintaining a continuous flow of energy and information with the environment. This provided the tools that served as the basis for generalization (for other fields) in his formulation of the ideas of GST [29].

Basically, the aim of GST is to formulate general principles and laws for systems, irrespective of their special features, the nature of their components and relations between them, a goal directly derived from the concept of open system [12].

## 4.2 FIR Methodology

The FIR Methodology is a subset of the GSPS methodology, located entirely at the hierarchical levels of the source, data and behavioral models. It deals with transformations (within each data and behavior levels) and transitions between the two levels.

As seen in Figure 4.1, FIR's cycle main functions are:

1. **Fuzzification**, which describes a transformation within the data model level, namely from the quantitative data model to its qualitative counterpart;
2. **Qualitative Modelling**, which describes the step up in the ladder from the data model to the behavioral model;
3. **Qualitative Simulation**, which denotes the transition back down the ladder to the previous level, and
4. **Defuzzification**, which performs another transformation at the data model level.

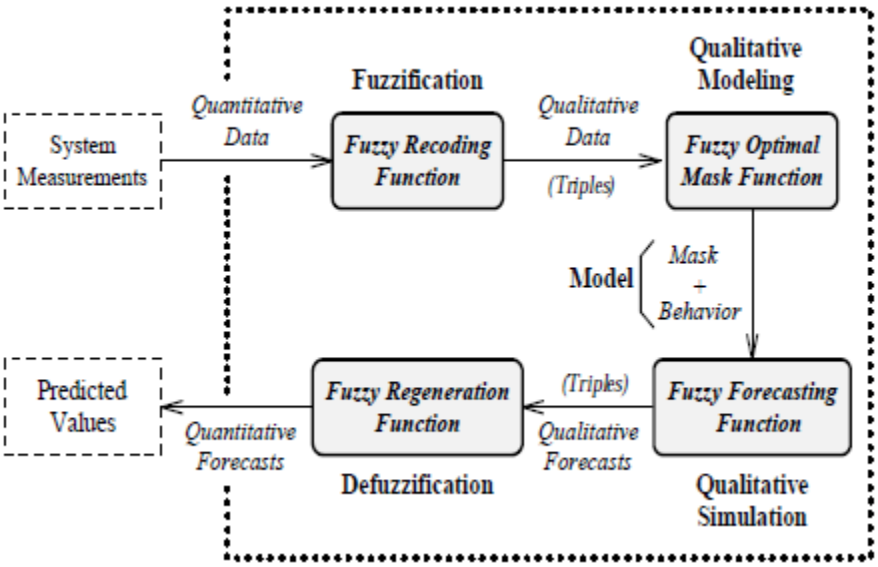


Figure 4.1. FIR process main stages.

- **Fuzzification**

FIR is fed with the available experimental data and the fuzzy recoding function converts quantitative values into qualitative triplets. The first element of the triplet is the class value, the second one is the fuzzy membership value and the third element is the side value.

The class value represents –roughly- a discretization of the original (real-valued) variable.

The fuzzy membership value denotes the level of confidence expressed in the class value chosen to represent a particular quantitative value.

The last element of the triplet, the side value, tells us whether the quantitative value is to the left/right/center of the peak value of the membership function. Although the side value is an unusual feature in FIR methodology, it is responsible for preserving the complete knowledge in the qualitative triplet that had been contained in the original quantitative value.

By converting the quantitative values to qualitative triplets, the search is drastically simplified, reducing the search space to the n-dimensional discrete search space of the class values. In most transformations some information is lost in the process; for example, an arithmetic temperature value contains more information than the values hot/warm/cold. This problem, though, is avoided using the fuzzy recoding technique.

- **Qualitative Modelling**

In the process of modeling, it is desired to discover causal relations among the variables that make the resulting state transition matrices as deterministic as possible. This is accomplished by means of the qualitative modelling function which is responsible for finding causal, spatial and temporal relations between variables that offer the best likelihood for being able to predict the future system behavior from its own past, thereby obtaining the best model (composed by the mask and the pattern rule base in the FIR terminology) that represents the system.

To establish qualitative relationships between different variables of the model is done using either exhaustive search in the discrete search space of the class values, or using Genetic algorithms reducing drastically the search time.

- **Qualitative Simulation**

Qualitative Simulation, or Fuzzy Simulation, is performed using the fuzzy forecasting function, which is able to predict future qualitative outputs –qualitative triples as described in the fuzzification step- from past similar experiences. This is done by interpolating

between previous instances of similar behavioral patterns, and uses these interpolated values to extrapolate the output variable. The FIR inference engine is based on a variant of the k-nearest neighbor rule. The forecast of the output variable is obtained as a weighted average of the potential conclusions that result from firing the k rules, whose antecedents best match the actual state.

- **Defuzzification**

Before the prediction is made, the final step is to implement the inverse process of the fuzzy recoding module -fuzzy regeneration function- most commonly known as the Defuzzification step. In essence this step converts qualitative triples back to quantitative values.

Most fuzzy logic signals lose information during fuzzification (first step) that cannot be retrieved in defuzzification. However, FIR's special feature is its fuzzy recoding, which preserves the complete information of the original –quantitative- value. This is done using an immediate cascade of a fuzzy recoding operation, followed by a fuzzy regeneration operation that restores the original values [12].

### 4.2.1 Fuzzification:

The first argument is a column vector containing the

MEMB_SHAPE: desired shape of the membership functions (bell-shaped/triangular)

$$n_{\mathrm{rec}} \geq 5 \cdot n_{\mathrm{leg}} = 5 \cdot \prod_{\forall i} k_i$$

Considering that in cluster analysis, each legal (all possible) discrete state should be recorded at least five times [12],

$n_{rec}$: total number of observed states - predetermined

$n_{reg}$: total number of distinct legal states

$n_{lev}$: optimum number of classes using the equation, assuming that all variables are classified into the same number of levels [12].

$$n_{\mathrm{lev}} = \mathrm{round}\ \left( \sqrt[n_{\mathrm{var}}]{n_{\mathrm{rec}}/5}\ \right)$$

$n_{var}$: number of input variables

At this point, the generated quantitative data have been converted into qualitative triples (Class, Membership, and Side). In addition, each one of the triples, stores the values in three matrices (of the same size) the “independent” matrices *Class/Memb/Side*.

Each column represents one of the observed variables and each row denotes one recording of all variables or one recorded state.

It is important no note, that although these parameters represent the optimal situation, and thus partitioning, in our case this could not be accomplished due to the reduced number of data points compared to the number of variables and their discretization.

### 4.2.1 Discretization Algorithms

In order to convert quantitative values to qualitative triples, it is necessary to specify the number of classes into which the definition domain of each variable is going to be divided, as well as the landmarks that separate neighboring classes from each other.

This is done in the second phase of the Visual FIR software -Recode-, and there are several algorithms to select from when doing a classification. For each attribute, a different algorithm can be selected and its parameters can be manually defined, such as number of clusters, number of iterations, stop criteria and several more depending on the selected algorithm. The list below includes the supported algorithms used and a brief description of their usage.

**Manual**

Here, we simply define the number of classes and the interval of each class.

**Single Linkage**

Single-linkage clustering is one of several methods of agglomerative hierarchical clustering. In the beginning of the process, each element is in a cluster of its own. The clusters are then sequentially combined into larger clusters, until all elements end up being in the same cluster. At each step, the two clusters separated by the shortest distance are combined.

**Complete Linkage**

Complete-linkage clustering is also one of several methods of agglomerative hierarchical clustering. The definition of "shortest distance" is what differentiates between the different agglomerative clustering methods. In complete-linkage clustering, the link between two clusters contains all element pairs, and the distance between clusters equals the distance between those two elements (one in each cluster) that are farthest away from each other. The method is also known as farthest neighbor clustering.

**Average Linkage**

The average linkage method (also a hierarchical clustering method) avoids the extremes of either large clusters or tight compact clusters. It is basically a compromise between the nearest and the farthest neighbor methods where the number of the elements of the new cluster can be predefined [14].

**Simple Average Linkage**

The simple average linkage method (mean linkage) takes both elements of the new cluster into account, using the sum of the two distances, multiplied by ½.

### Centroid Linkage

The centroid is defined as the center of a cloud of points and centroid linkage techniques attempt to determine the “center” of the cluster. With the centroid linkage method, the distance between two clusters is the distance between the cluster centroids or means. Like the average linkage method, this method is one more averaging technique.

### Median Linkage

This method is similar to the previous one. If the sizes of two groups are very different, then the centroid of the new group will be very close to that of the larger group and may remain within that group. This is the disadvantage of the centroid method. It could be made suitable for both similarity and distance measures as it takes into consideration the size of a cluster, rather than a simple mean.

### Ward Linkage

The main difference between this method and the linkage methods is in the unification procedure. This method does not join groups with the smallest distance, but it rather joins groups that do not increase a given measure of heterogeneity by too much. The aim of Ward’s method is to unify the groups such that variation inside these groups does not increase too drastically. This results in clusters that are as homogenous as possible [30].

### Equal Width Interval

Due to its mere simplicity, equal width interval binning is very popular and usually implemented in practice. The algorithm needs to first sort the attribute according to its values, and then find the minimum and maximum value of that attribute in order to compute the interval width.

### K-Means

K-means is one of the simplest unsupervised learning algorithms. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters

(assume k clusters) fixed a priori. The main idea is to define *k* centroids, one for each cluster.

**Hard C-Means**

In the classical 'hard' c-means clustering algorithm, each data point is a member of one and only one cluster and the number of clusters, *c*, necessary to correctly cluster the data is known a priori.

**Fuzzy C-Means**

Fuzzy c-means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method developed by Dunn in 1972 [33], to be improved later on by Bezdek in 1981 [34], frequently used in pattern recognition.

**Equal Frequency Partition (EFP)**

EFP is sensitive to data distribution, since it partinions the data into parts of equal length. A good partitioning is obtained if all possible behaviors of the system are represented with a comparable number of occurrences.

**Enhanced Equal Frequency Partition (EEFP)**

The EEFP method eliminates multiple observations of the same behavioral pattern.

It achieves that by having 2 parameters (δ and a) where,

δ = range of similar observations.

α = minimum number of occurrences to assume that this behavioral pattern is over-represented.

Below, Figure 4.2 shows a screenshot of the interface in which all the steps are performed.

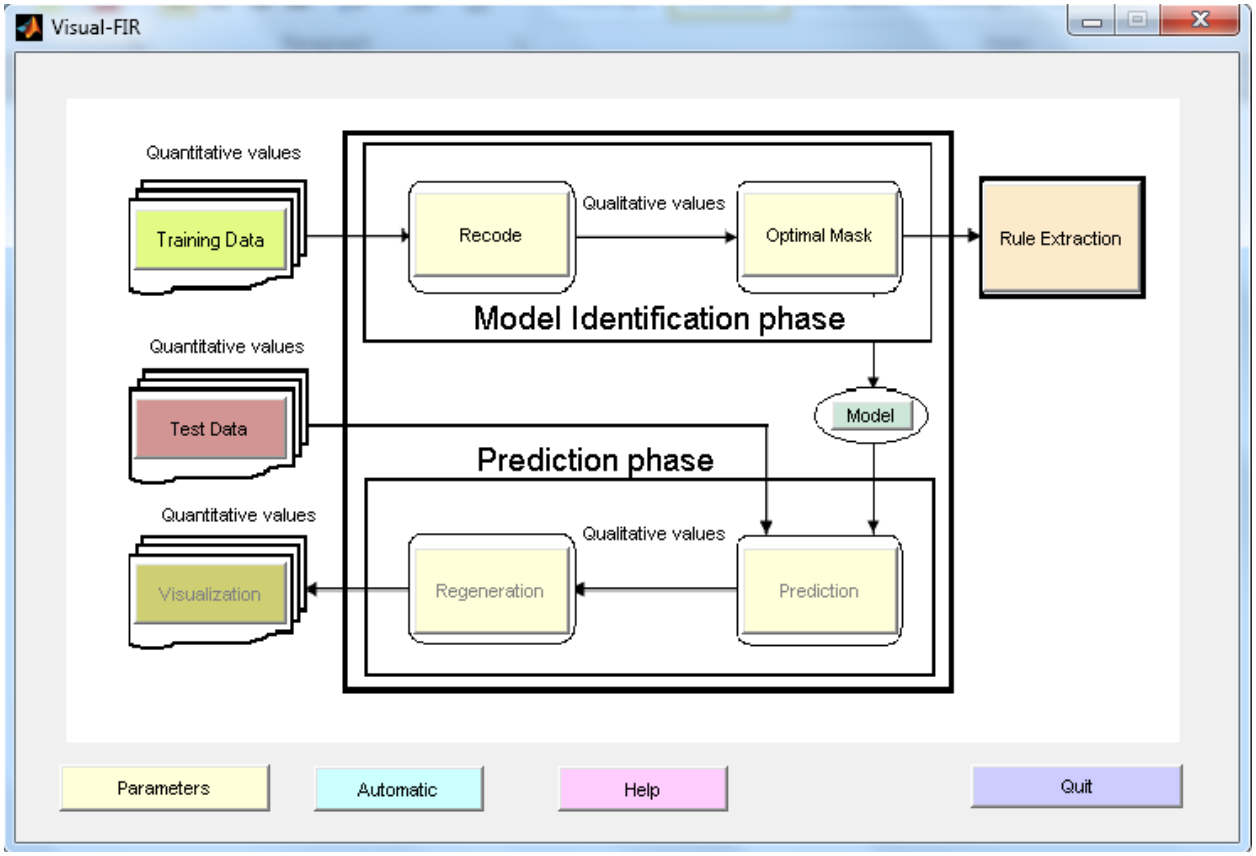


Figure 4.2. VisualFIR's software main window

Figure 4.3 shows the fuzzification step (the Recode step in the previous interface -Figure 4.2-) where we can select the Classification algorithm, the clusters per variable as well as parameter fine-tuning depending on the algorithm selected.

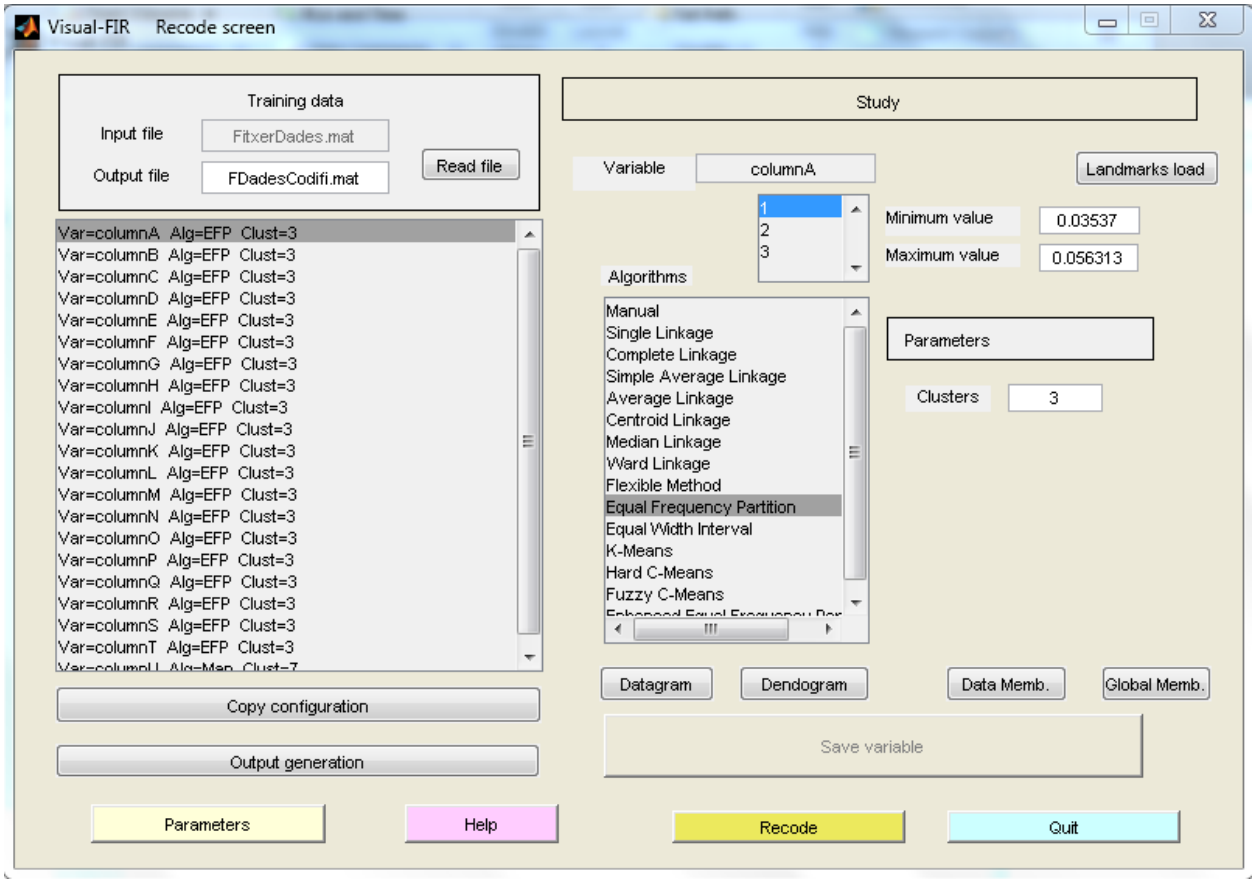


Figure 4.3. Classification selection in VisualFIR software

### 4.2.2 Qualitative Modelling

In the FIR methodology, the Fuzzy Modelling process is performed using the Optimal Mask function that optimizes the predictiveness of the model by performing a search in the discrete space of the Class values.

The real inputs and a set of measurable outputs are recorded as functions of time, stored in a matrix where can be separated into a set of inputs ($u_i$) concatenated with a set of outputs ($y_i$) [12].

We want to find such finite automata relations among the recoded variables that will make the resulting state transition matrices as deterministic as possible and if such relationship is found for every output variable, the system's behavior can be forecasted by iterating

through these matrices; the more deterministic the matrices, the higher likelihood that the future system behavior will be predicted accordingly.

In FIR, a mask denotes a dynamic relationship among qualitative variables, that has the same number of columns as the inputs and the mask's rows are equal to the desired mask depth.

It is important to note that in our case the data are not time-related so we will be using a mask depth of 1.

Our mask will eventually be a vector (1 x *n* matrix) containing negative values that denote input arguments of the qualitative functional relationship (mask input) and are the "holes" of the mask that are visible when the mask is in a specific position. Values equal to 0 are when no relationship is found and the single (last) positive value denotes the output.

After the mask has been applied, the –former dynamic behavior- becomes stored within single rows.

As mentioned, the best mask is chosen by performing exhaustive search through all legal masks of complexities 2, 3, … , 9. The quality of the mask will grow with increasing complexity, then reach a maximum and decrease rapidly. A good value for the maximum complexity is usually 5 or 6 [12].

Each of the possible masks is compared to the others with respect to the maximization of its forecasting power using overall entropy reduction.

The FIR optimization formula uses the Shannon entropy measure:

$$H_i = \sum_{\forall o} \mathrm{p}(o|i) \cdot \log_2 \mathrm{p}(o|i)$$

,

used to determine the uncertainty associated with forecasting a particular output state, related to one input state where $p(o|i)$ is the conditional probability of a certain mask output state o to occur, given that the mask input space has already occurred. The probability here

stands as the division of the observed frequency of a particular state, divided by the highest possible frequency of that state.

The sum of the overall entropy of the mask is found:

$$H_m = -\sum_{\forall i} \mathrm{p}(i) \cdot H_i$$

Where $p(i)$ is the probability of that input state to occur. When all probabilities are equal, the highest possible entropy ($H_{max}$) can be obtained.

Normalized overall entropy reduction formula becomes:

$$H_r = 1.0 - \frac{H_m}{H_{\max}}$$

where $H_r$ would be a real number in the range between 0.0 – 1.0, where higher values usually indicate an improved forecasting power, as the masks with highest entropy reduction values generate forecasts with the smallest amounts of uncertainty.

We have a class input/output matrix, where the confidence of the row of this matrix is the joint membership of all the variables associated with that row. A confidence vector indicates how much confidence can be expressed in the individual rows of the class input/output matrix.

Computing these two we can get the basic input/output behavior of the model as an ordered set of all observed (distinct) states, along with a measure of confidence for each state. The individual confidence of each state is accumulated, and if a state has been observed more than once, more confidence can be expressed in that state. The cumulative membership is the addition of the individual confidences of a given state (all occurrences of the same input). This is decided due to the best mask selections in a fairly large number of experiments using the FIR methodology [12].

Higher complexity masks with a large entropy reduction value but smaller overall quality-due to high complexity- will usually provide excellent forecasts but they will probably not be able to produce forecasts at all. Since the total number of observed states remains constant, the frequency of observation of each state shrinks rapidly; and so does the predictiveness of the model-with increasing complexity, *Hr* simply keeps growing-. Thus, a situation where every state that has been observed has been observed precisely arises which leads to $H_r$ assuming the maximum value of 1.0. This problem is overcome using an observation ratio, $O_r$, introduced based that from a statistical point of view, every state should be observed at least five times.

$$O_r = \frac{5 \cdot n_{5\times} + 4 \cdot n_{4\times} + 3 \cdot n_{3\times} + 2 \cdot n_{2\times} + n_{1\times}}{5 \cdot n_{\text{leg}}}$$

where,

$n_{leg}$: number of legal mask input states

$n_{1x}$: number of mask input states observed only once

$n_{2x}$: number of mask input states observed twice

$n_{3x}$: number of mask input states observed thrice

$n_{4x}$: number of mask input states observed four times

$n_{5x}$: number of mask input states observed five times or more

Now, $O_r$ can be used as a quality measure bearing in mind that if every mask input state has been observed at least five times will be equal to 1.0. The optimal mask is the mask with the largest $Q_m$ value.

$$Q_m = H_r \cdot O_r$$

It is important to point once again, that the observation ratio does not influence the quality of a forecast. It simply influences the likelihood that a forecast can indeed be made.

Figure 4.4 shows the Optimal mask screen, where we define our desired mask complexity, the mask depth as well as the search algorithm (Exhaustive/Genetic). Furthermore, the user gets an estimated computation time, and when finished, the mask as well as its quality is shown as below.

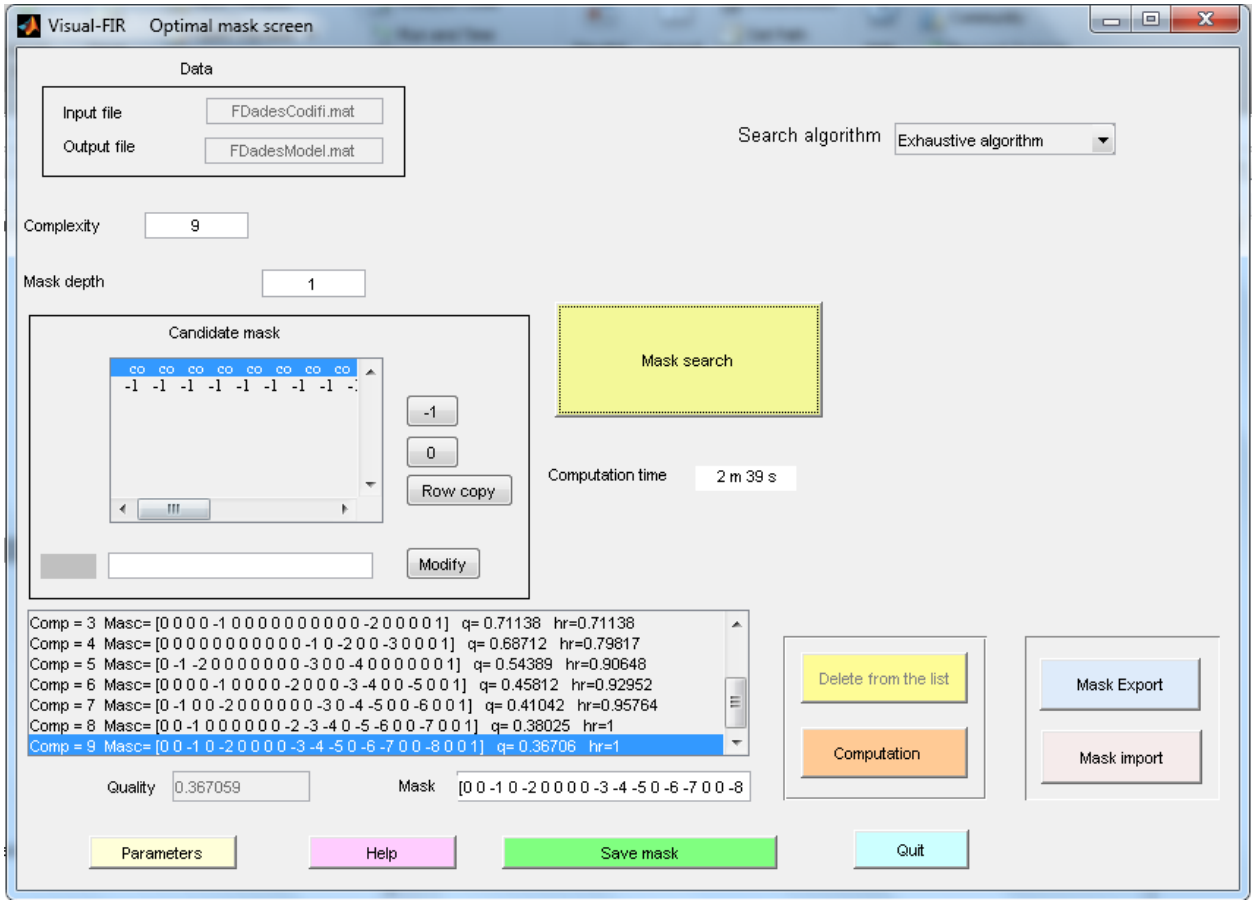


Figure 4.4. Mask Computation in VisualFIR software.

Once we apply the mask to the qualitative model, we obtain what is known as a “static episodical behavior”, where the static or pattern rules can be interpreted as a kind of rule base.

### 4.2.3 Qualitative Simulation

Once an optimal mask has been determined, we can obtain the class input/output matrix, as well as the membership and side value matrices, since the matrices contain functional relationships within single rows. The rows are then sorted in alphanumerical order and we obtain the behavior matrices. These matrices are composed by the class, membership and a side behavior matrix; the latter bring something like a finite state machine; for each input it shows which output is most likely to be observed.

In fuzzy forecasting, it is essential that the class value of the output as well as the fuzzy membership and the side values are forecast, so we can predict an entire qualitative triple from which a quantitative variable can be regenerated whenever needed.

The fuzzy forecasting method used in these experiments, has generated more accurate forecasts than alternative methodologies such as the Center of Area (COA) or Mean of Maxima (MOM) in most past experiments [12].

The membership and side functions of the new input state are compared with those of all previous recordings of the same input state contained in the class behavior matrix and a normalization function (selection between two functions) is computed for every element of the new input space. The normalization function is basically a transformation from the qualitative triple to the quantitative variable (differs than the original quantitative variable).

**Normalization function 1:**

$$p_i = Side_i \cdot B \cdot \sqrt{\ln Memb_i} + 0.5$$

where $B = (4 \ln 0.5)^{-1/2}$

Left class:

$$p_i = C \cdot \sqrt{\ln Memb_i}$$

Right class:

$$p_i = 1 - C \cdot \sqrt{\ln Memb_i}$$

where $C = (\ln 0.5)^{-1/2}$

Irrespective of the original signal, the $p_i$ signal ranges between 0.0 – 1.0 [12].

**Normalization function 2:**

$$p_i = Class_i + Side_i \cdot (1.0 - Memb_i)$$

$p_i$ are the quantitative (normalized) variables that represent the relative magnitude if the original qualitative triple. In this case, the $p_i$ ranges between 1.0 – 1.5 for the lowest class and between 1.5 – 2.5 for the next higher class, etc.

It is important to note that since $p_i$ values are bot regenerations of the original quantitative triple, different $p_i$ signals can be compared or summed up, without weighing them relative to each other; something that would not be meaningful using the original or regenerated values. We then get the norm image (vector $p$) of the original input space.

$p = [p_1, p_2, \ldots, p_j]$,

assuming that the state contains $j$ mask inputs.

We get norm images for every previous recording (of the same input space) that are stored in the $p_k$ vectors. These vectors have different membership and side function values, but identical recorded input states.

Computation of $L_2$ norms of differences between $p$ and $p_k$ vectors representing all previous recordings of the same input space:

$$d_k = \|\mathbf{p} - \mathbf{p_k}\|_2 = \sqrt{\sum_{i=1}^{N} (P_i - P_{ik})^2}$$

We now obtain a set of the *k* elements that have the smallest $L_2$ norm (if they are found in the class behavior matrix) and they are used to forecast the new output space. These set of elements is also called the *k nearest neighbors*, where each neighbor has a distance/weight to each neighbor as a function of proximity.

What we want is that if one of the previous observations leads to a very small distance function, its weight should "dominate" the computation and if all distance functions are equally large, we should make use of an arithmetic mean between the previous distance functions. Different formulas that are enabled using a global variable (ABS_WEIGHT) are used to deal with this issue.

For example, if none of the five smallest distance functions, $d_k$ is equal to zero, we use:

$$w_{\mathrm{abs}_k} = \frac{(d_{\max}^2 - d_k^2)}{d_{\max} \cdot d_k}$$

$k$ : loop over the five nearest neighbors

$d_i <= d_j$ , $i<j$, $d_{max} = d_5$

However, if any of the $d_k$ values is zero, the formula needs to be modified to deal with this, where we use the following equation:

$$w_{\mathrm{abs}_k} = \begin{cases} 0.0 \; ; & d_k \neq 0.0 \\ 1.0 \; ; & d_k = 0.0 \end{cases}$$

After computing the absolute weights, using the sum of the five absolute weights we can compute the relative weights that can be interpreted as percentages, so that the output state values can be computed as a weighted sum of the output states-of the previously observed k nearest neighbors- where a qualitative output space can be computed.

### 4.2.4 Defuzzification

In this stage, a function is used that implements the inverse process to the recode function that uses the same equation for recoding/regenerating values. The defuzzification part (as well as the fuzzification part) is not part of the reasoning process, but essential to enable us to operate in a mixed quantitative/qualitative modelling and simulation environment.

We have a class vector ($f_c$) membership vector ($f_m$) and the side vector ($fs$) that contain the qualitative forecast and are converted back into a quantitative trajectory, $r$. Here, as in the fuzzification part, we can also choose between gaussian and triangular membership shapes (using the global variable MEMB_SHAPE).

In the experiments done -regarding class prediction- we are interested in classification and not regression. In this case, we do not perform the next step regarding regeneration; instead we focus in the class predicted values.

Now that all the steps in the FIR methodology have been clarified, Figure 4.5 shows the principal data structures along all processes.

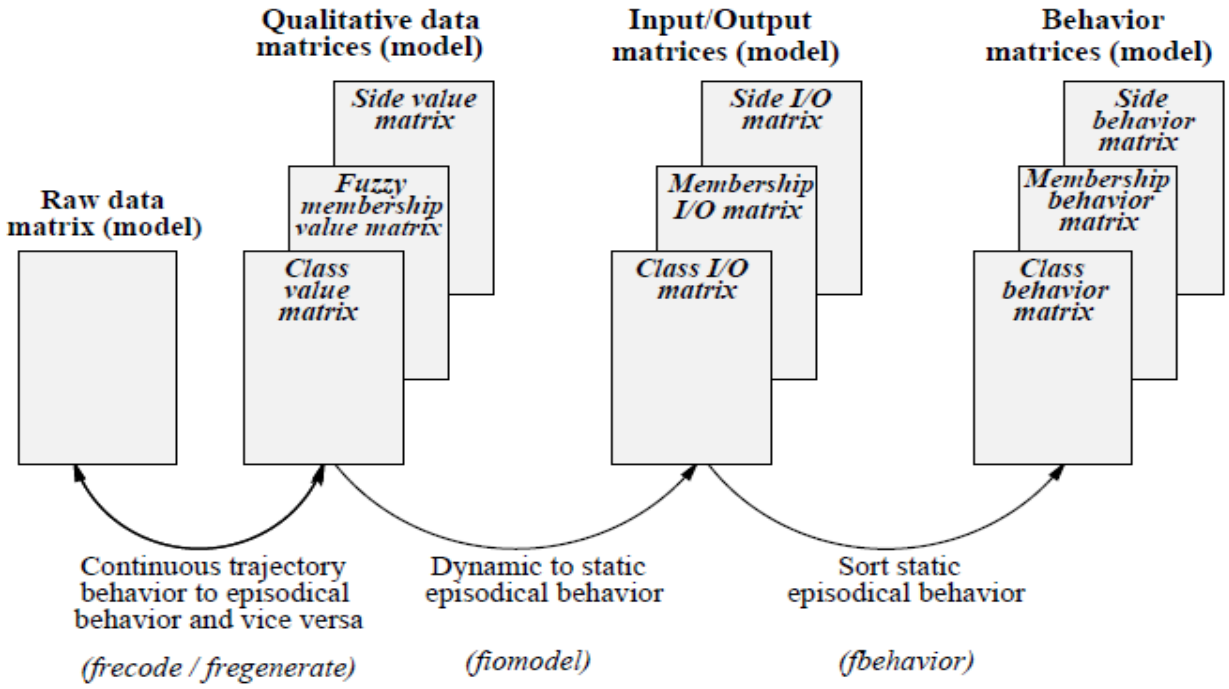


Figure 4.5. Main FIR data structures and relationships

When this data is fuzziffied by means of the *recode* function the qualitative data matrices (model) are obtained. The qualitative data model is composed of three matrices, one with the class values, the second with the fuzzy membership values, and the third with the side values. The next step that affects system's data, converts the dynamic episodical behavior into a static episodical behavior (*iomodel* function). This process is described also in Figure 4.5. From the qualitative data matrices the input/output model is obtained, composed by the class I/O matrix, the membership I/O matrix and the side I/O matrix. These matrices contain already the "rules" that describe the system. The input/output matrices are sorted in alphanumerical order when the *behavior* function is used. The new sorted matrices are called Behavior matrices (model). The first matrix contains the class behavior, the second one the membership behavior and the last one the side behavior. The behavior model (matrices) is used inside the inference engine during the prediction process, as has been explained in the section entitled "qualitative simulation".

### 4.3 Linguistic Rule Extraction

One of the potential drawbacks affecting the application of CI methods in general to the analysis of data is the often limited interpretability of the results they yield. One way to overcome interpretability limitations is by explaining the operation of CI models using rule extraction methods.

Reasoning with logical rules is more acceptable than the recommendations given by black box systems, because such reasoning is comprehensible, provides explanations, and may be validated by human inspection. It also increases confidence in the system, and may help to discover important relationships and combination of features, if the expressive power of rules is sufficient for that [57].

#### 4.3.1 LR-FIR Methodology

Linguistic Rules (LR) in FIR is a novel rule-extraction algorithm based on fuzzy logic that starts from the FIR methodology.

The proposed algorithm, LR-FIR, is able to derive linguistic rules from a FIR model, to obtain good qualitative relationships between the variables that shape the system and to predict the future behavior of that system.

The main trait of LR-FIR is that it is able to compact the pattern rule base obtained by FIR into a much reduced set of linguistic rules, which contains the main aspects of system's behavior. This is shown in the LR-FIR structure box represented in Figure 4.6.

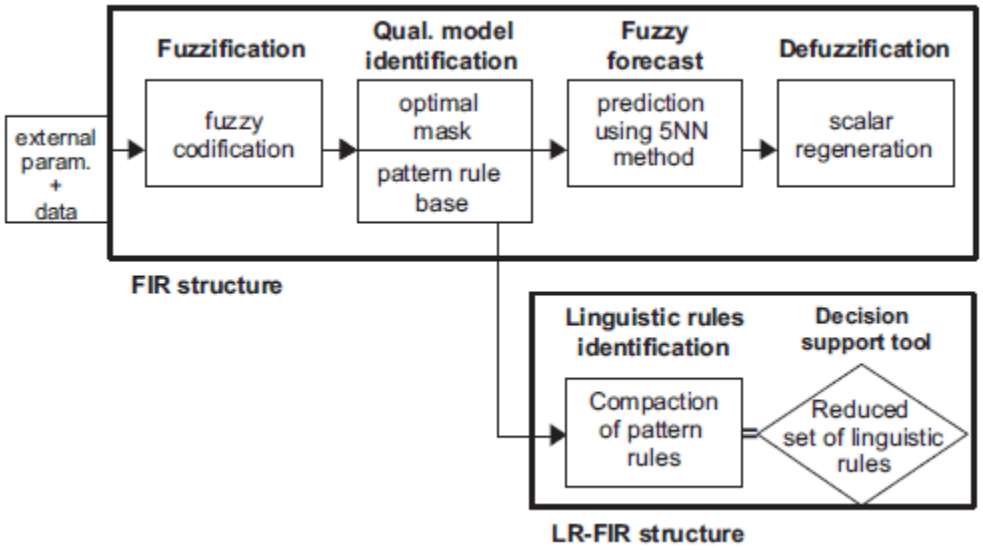


Figure 4.6. FIR and LR-FIR main structures [57].

However, the premises and consequences of rules are not necessarily binary in nature and, therefore, the algorithm must be able to deal with multi-valued logic, and accept partial *do-not-care* conditions. Due to the fact that LR-FIR was developed within the FIR methodology, the obtained rules could be considered as predictive rules and deal naturally with the uncertainty captured in the FIR models.

LR-FIR extracts predictive rules of the IF–THEN type and allows representing multi-valued logic functions, i.e. neither inputs nor outputs are restricted to binary logic. Instead of avoiding overlapping rules, these rules are treated in the compaction and unification steps where rules sharing contiguous input spaces in a feature and the same values in the remaining features are unified in a unique rule. In this way, LR-FIR represents as

accurately as possible the system behavior while preserving the main goal, i.e., the simplicity of the resulting rule base.

The main phases of the LR-FIR algorithm are shown in Figure 4.7. The algorithm performs an iterative process that compacts the pattern input/output relationships obtained by FIR in order to obtain interpretable, realistic and efficient rules describing the behavior of the analyzed system. The premise of rules is limited to a conjunction of attributes. Disjunction is not supported in LR-FIR; however, the presence of multiple rules with the same conclusion could represent disjunction.

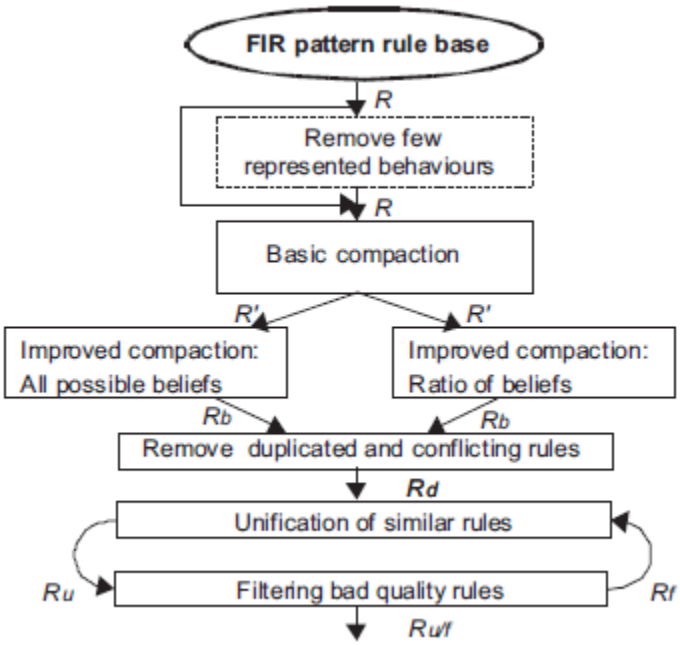


Figure 4.7. Main steps of the LR-FIR algorithm [57]

**Basic compaction**

LR-FIR is dependent on the number of variables and pattern rules, due to the fact that it is an iterative process that evaluates, one at a time, all the variables in a rule and all the rules in the pattern rule base. Therefore, the larger the number of variables and the pattern rules

involved is, the higher the computational time needed for the compaction of pattern rules into linguistic rules.

The basic compaction step is an iterative step that evaluates, for each variable or premise, all the rules in the pattern rule base ($R$). $R$ is compacted on the basis of the "knowledge" obtained by FIR. A subset of rules ($R_c$) can be compacted in the form of a single rule $r_c$, when all premises $P$ but one ($P_a$), as well as the consequence $C$, share the same values. Premises, in this context, represent the input features, whereas consequence is the output feature in a rule. If the subset contains all legal values $L_{Va}$ of $P_a$, all these rules can be replaced by a single rule, $r_c$, that has a value of −1 in the premise $P_a$. A value of −1 in a variable means that the variable is not relevant in the rule and, therefore, it is not considered [57].

It is important to note that the algorithm performs two iterations: an external one that deals with each one of the premises and an internal one that deals with each rule of the pattern rule base. These are some considerations about this algorithm:

1. When more than one −1 value is present in a compacted rule $r_c$, it is compulsory to evaluate the existence of conflicts by expanding all the premises to all their legal values $L_{Va}$, and comparing the resultant rules with the original pattern rules $R$. If conflicts exist, the compacted rule $r_c$ is rejected, and otherwise accepted. In the latter case, the previous subset, $R_c$ is replaced by the compacted one $r_c$. Conflicts occur when one or more extended rules have the same values in all its premises but different values in the consequence.
2. When a value of −1 appears in any of the variables that are not the one evaluated at this moment ($v_j$), the −1 includes the class values of the other rules in that premise. An early unpublished version of this algorithm was proposed by F.E. Cellier and S. Medina in 1997 [57].

**Improved Compaction**

- **Improved compaction 1**

Whereas the previous step only structures the available knowledge and represents it in a more compact form, the improved compaction step extends the knowledge base *R* to cases that have not been previously used to build the model, $R_b$. Thus, whereas the basic compaction algorithm leads to a compressed data base that only contains knowledge, the improve compaction algorithm contains undisputed knowledge and uncontested belief.

Two options are studied: all possible beliefs (APB) and ratio of beliefs ($R_B$). In the first option, improved compaction with all possible beliefs, starting from the compacted rule base *R*' (refer to Fig. 4.7), all premises *P* are visited once more, in all the rules, *r*, that have non negative values, and their values are replaced by −1 elements. Then, for each −1 value studied, an expansion to all possible full set of rules and their comparison with the original rule base *R* are carried out. If no conflict $C_f$ results, the new rule, $r_c$, is accepted, and, otherwise, rejected. Notice that the test for conflicts is done for any compacted rule, as has been done in the basic compaction. Remember that conflicts occur when one or more extended rules have the same values in all its premises but different values in the consequence.

- **Improved Compaction 2**

In the case of a complete knowledge base, the enhancement is always harmless. In the case of an incomplete knowledge base, it may not be. For this reason, a second more conservative algorithm is introduced. The improved compaction with a ratio of beliefs algorithm works as the previous one but it is more demanding. The candidate rule is accepted if and only if no conflict results and a ratio, $R_A$, of the set of expanded rules obtained from the candidate rule is found in the original rule base *R*, i.e. the $R_A$ refers to how many of the instances of the set of expanded rules exist in the original rule base *R*.

Although a ratio of beliefs is used to compact $R_c$ in a single rule $r_c$, it is minimal and does not compromise the model previously identified by FIR. This latter option is, therefore, more

conservative than the first improved compaction algorithm, where beliefs are assumed to be consistent with the original rules.

**Remove Duplicates and Conflicting Rules**

In the previous steps, i.e. basic compaction and improved compaction, no conflicting rules can be produced. The existent conflicting rules come from the pattern rule base that is the starting point of the LR-FIR algorithm.

The reason of the existence of conflicting rules in the pattern rule base is clear. As mentioned in Section 4.2.1, the fuzzification process of FIR methodology converts quantitative data into fuzzy data that consist of the class and the fuzzy membership values. The class value represents a coarse discretization of the original real-valued variable. The fuzzy membership value denotes the level of confidence expressed in the class value chosen to represent a particular quantitative value. Therefore, in the fuzzy world, two or more rules that have the same antecedents and different consequent are not necessarily ambiguous or conflicting rules, because each one has a membership grade associated to that consequent. The conflicting rules appear when we move from the fuzzy world to the classical Boolean world.

In this step of the LR-FIR algorithm, the rules that are involved in conflicts are analyzed, and those with lower quality, $Q_r$, are eliminated. The quality of a rule is assessed using the well-known specificity and sensitivity measures that are standard metrics often applied in the machine learning field. Specificity and sensitivity measures are in the range [0–1]. High quality means high values (closer to 1) of specificity and sensitivity measures; therefore the rule with highest $Q_r$ remains and the other conflicting rules are eliminated. In order to maintain a robust and consistent set of rules, those conflicting rules sharing contiguous input space (adjacent classes) in the consequent are not removed since these rules should be unified in the next step of the algorithm.

**Rules Filtering**

The obtained set of rules $R_d$ or $R_u$ is evaluated using the sensitivity and specificity metrics that are defined in a following (Rules Evaluation) section. These metrics allow an objective

and realistic assessment of the resulting rules. A parameter, chosen by the user, determines the minimum quality value to be accepted. The rules that have lower qualities associated for, at least, one of the metrics are eliminated.

**Rules Unification**

Performing the unification step after the Rules Filtering step, low quality rules that have conflicts with better quality rules are eliminated and therefore, the resulting set of rules explain in a more synthesized and clear way the more often behaviors of the system. Although some information is obviously lost, the set of rules obtained is usually more suitable for decision support. LR-FIR offers the choice of selecting the order. If it is done before, rules with low quality are usually unified with rules of better quality deriving, more often than not, to a set of rules that preserve as much as possible the full behavior of the system but are not really useful for decision support systems.

Rule unification is an iterative process that evaluates, one at a time, each rule with respect to the remaining ones to find similar candidate rules to be unified in a single one $R_u$. This is carried out in two phases. In the first phase, the rules that share, in a same variable (premise or consequent), contiguous input spaces and the same values in the remaining ones, should be unified in a unique rule $R_u$.

In order to maintain a consistent set of rules and do not compromise the previous steps, a subset of rules, *Rd*, should not be unified when the contiguous input space of the candidate rules $R_d$, cover all the legal values of that variable. This condition is included because when this happens a conflict surely exist, otherwise these candidate rules $R_d$, would have been compacted in the previous -basic or improved compaction steps-. There are four options to perform this step:

- *Wise*: A subset of rules $R_d$ is unified in a unique rule $R_u$, if and only if the quality $Q_r$, of the unified rule $R_u$, is higher than the best quality of the candidate rules;
- *Blind*: a subset of candidate rules $R_d$ is unified without verifying the quality $Q_r$ of the unified rule $R_u$.
- These two alternatives can be combined with repetitions, and

- without repetitions options.

In the first one, a rule can be unified with several rules, i.e. whenever possible, whereas in the second one, a rule can be unified only once. Optionally, those rules not unified in the first stage are evaluated with the goal to discover new unifications with the already unified rules. The unification is performed only in the consequent value. The default option is without repetitions, a rule can be unified only once.

**Rules Visualization**

This is the final step of the rules extraction process in the LR-FIR methodology, where the rules are visualized in a readable format of conjunctions between the input variables in respect with the related class output.

**Rules Evaluation**

The rules assessment is performed using the original system's data (used by FIR methodology to obtain the FIR model). As explained before, each rule is evaluated using the sensitivity and specificity metrics, both based on the well-known confusion matrix (TP, FN, FP and TN).

Sensitivity is the ratio of the number of in-class data that the rule identifies to the total number of in-class data. Specificity is defined as the ratio of the number of out-of-class data that the rule identifies to the total number of out-of-class data. Both metrics are in the range [0–1]. It is desirable that both the specificity and the sensitivity reach high values close to value 1. A high sensitivity value implies a very general rule, i.e. a high number of data points fit in that rule. A small sensitivity value denotes a very specific rule, i.e. the rule embraces a small set of data points. The chosen metrics allow an objective and realistic assessment of the resulting rules, independently of the data distribution of each class. The formulae to calculate the specificity and sensitivity, metrics can be written in several forms:

$$sensitivity = \frac{TP}{TP + FN} = \frac{TP}{\text{Tot-in-class}} = 1 - \frac{FN}{TP + FN}$$

$$specificity = \frac{TN}{TN + FP} = \frac{TN}{\text{Tot-out-of-class}} = 1 - \frac{FP}{TN + FP}$$

- TP (“true positive”) is the number of cases that the rule predicts that fit in the class x, and really belong to the class x.
- FN (“false negative”) is the number of cases that the rule predicts that not fit in the class x, and really belong to the class x.
- FP (“false positive”’) is the number of cases that the rule predicts that fit in the class x, and really not belong to the class x.
- TN (“true negative”) is the number of cases that the rule predicts that not fit in the class x, and really not belong to the class x.

Tot-in-class denotes the total number of real data that fit in the actual class.

Tot-out-of-class denotes the total number of real data that do not fit in the actual class.

In order to provide an integral and consistent evaluation of the rules, LR-FIR also gives a joint evaluation metrics for each class of the output attribute.

Below, in Figure 4.8, the VisualFIR interface of the rule extraction process in which all the above steps are performed is shown. The Rules illustrated are irrelevant at this point.

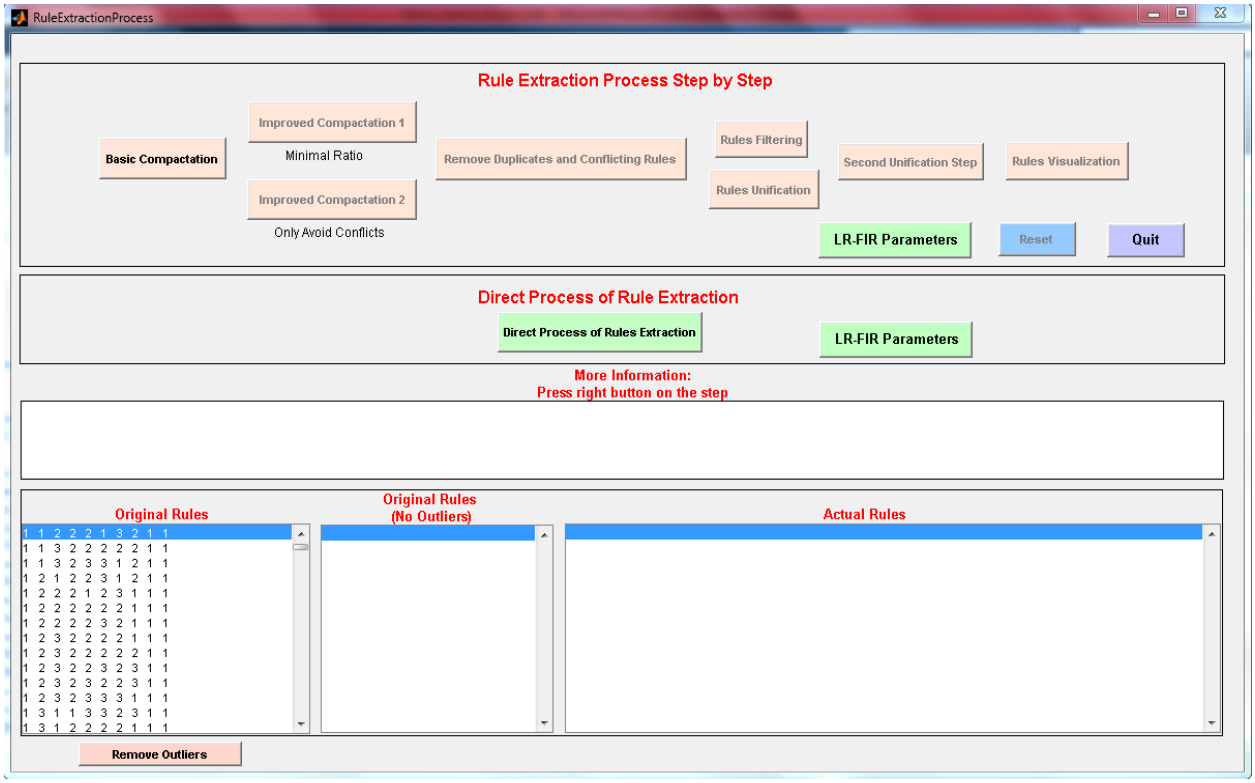


Figure 4.8. An illustration of the process of Rule Extraction using VisualFIR software.

### 4.3.2 Rule Extraction in biomedical applications

As discussed in the previous section rule extraction processes are of extreme interest. Several CI techniques have been recently proposed for the prediction of protein structure, but statistical methods are mostly based on the likelihood of each AA sequence being one of three types of secondary structures [51] [52]. Many machine learning techniques such as SVMs largely function as a black box, thus often not being interpretable in regard to the predictions they make.

Limited studies focus on the discovery of logic rules underlying the prediction itself. Such approaches have been used on the extraction of rules from protein secondary structures (PSS), resulting in usually few and compact rules, with strong support by biological evidence [53].

Another approach regarding protein molecular structures, cohesive structural itemset mining, has been used to extract patterns of amino acids in spatial proximity -within a set of proteins- based on their atomic coordinates in the protein molecular structure. The patterns extracted seem to reflect AAs with a supporting role to the overall or specific structure of the protein [54].

A combination of SVM and Decision Trees has been used to add the SVM's strong generalization ability to the comprehensibility of rule induction. Specifically, SVM has been used as a pre-processing step of the Decision Tree, resulting in rules that have strong biological meaning [55].

Finally, rule extraction methods have also been used to successfully extract information about protein interactions from scientific literature, employing only a protein name dictionary, and achieving high recall and precision rates for yeast and *escherichia coli* [56].

# Chapter 5

# Experimental Work: Results and Discussion

We run our experiments using the VisualFIR software, a tool for model identification and prediction of dynamical complex systems, implemented using Matlab code in release 2012b.

In the following sections we will discuss the experiments done regarding the linguistic rule extraction process, using the LR-FIR software included in the VisualFIR, as well as a classification approach, including an approach proposed to improve on the classification results.

Section 5.1 present the linguistic rules obtained using the data sets of Class C GPCR and the mGluR, in respectively.

Section 5.2 follows the classification part, regarding the discretization of the input variables, and thus, the extraction of optimal masks based on the Fuzzification and Qualitative Modelling part of the FIR methodology, on the Class C GPCR dataset.

Finally, section 5.3 discusses on the results obtained in the previous sections.

## 5.1 Rule Extraction

### 5.1.1 Rule Extraction for Class C GPCR data set

As mentioned in the previous chapter, the Qualitative Modelling step returns a set of pattern rules, which are used by the LR-FIR to extract the linguistic rules.

The rules in the current experiments were extracted using the VisualFIR user interface software, as illustrated in Section 4.3.1 - Figure 4.8. The whole set of rules is included in the Appendix section at the end of the thesis document, for it not to interfere excessively with the comment of the main results.

The format of the rules to be extracted is based on the:

- Fuzzification part,

  In this part, the partition of the input and output variables is defined. The inputs, which are, basically, the frequencies of apparition of the 20 AA in the sequence alphabet, are divided, in our case, into three partitions or ranges of values, namely Low / Medium / High, based on their values. The output variable, namely CLASS, is accordingly partitioned into the seven output classes (the seven subtypes of the class C GPCR superfamily).

  A study of different discretization algorithms was performed, in order to select the one with the best performance, to be used for the extraction of pattern rules; and in turn the linguistic rules.

- Qualitative modelling part,

In this part, we use the mask of Complexity 9 (highest complexity currently supported by VisualFIR) obtained in the experiments. This translates into nine selected attributes -eight input variables plus the output- for the description of the rules.

The mask extracted for the Class C GPCR dataset is the following:

**[ -1 -2 0 0 0 -3 0 -4 0 -5 -6 0 0 -7 0 0 0 0 0 -8 1]**

The negative values in the columns represent the input variables selected by the FIR steps to be *the most relevant ones*. The zero values correspond to the input variables not selected, and the positive value of 1 just identifies the output variable.

The relevant AAs according to the extracted mask are highlighted in Table 5.5.

For more information, the reader is referred to the full table of the AAs, as presented in Chapter 3 (Table 3.1).

| -1 | -2 | 0 | 0 | 0 | -3 | 0 | -4 | 0 | -5 | -6 | 0 | 0 | -7 | 0 | 0 | 0 | 0 | 0 | -8 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | R | N | D | C | Q | E | G | H | I | L | K | M | F | P | S | T | W | Y | V |

Table 5.5 Correlation of mask values and input variables, regarding the AA alphabet.

A sample of the rules extracted is shown in Table 5.6 for illustration, with a single rule for each output class.

| RULES |
|---|
| IF R IN3 AND G IN 3 AND L IN 1 AND V IN 3 THEN CLASS IN 1 |
| IF Q IN 3 AND I IN 2 AND L IN 1 AND F IN 3 AND V IN 1 THEN CLASS IN 2 |
| IF A IN 3 AND Q IN 3 AND L IN 2 AND F IN 1 AND V IN 3 THEN CLASS IN 3 |
| IF Q IN 3 AND G IN 2 AND L IN 3 AND F IN 3 AND V IN 3 THEN CLASS IN 4 |
| IF A IN 1 AND Q IN 1 AND G IN 2 AND L IN 2 AND F IN 2 AND V IN 1 THEN CLASS IN 5 |
| IF Q IN 2 AND G IN 2 AND I IN 2 AND L IN 3 AND F IN 3 AND V IN 2 THEN CLASS IN 6 |
| IF R IN 1 AND Q IN 3 AND G IN 2 AND I IN 1 AND L IN 3 AND F IN 2 AND V IN 3 THEN CLASS IN 7 |

Table 5.6. Sample of Linguistic Rules for Class C GPCR

Rules are expressed in an IF – THEN format. In between, the input variable/s and their corresponding range are described. If there is more than one input variable considered in the rule, this is presented in the form of conjunctions (*AND*).

The final part of the rule, a consequent *THEN,* is the assignment of a sequence that complies with the rule(s) to a given class C GPCR subtype (CLASS).

As we mentioned above, each capital letter corresponds to an amino-acid (A-V as shown in Table 5.5) that is partitioned in three subsets.

Numbers *1-3* correspond to the values of Low / Medium / High for an individual input (AA).

The rule part, *R **IN** 3*, for example, means that the value of the Q is high in the specific -first- rule.

In the following seven tables (Tables 5.7-5.13), one for each output class, we have compactly summarized the number of appearances of each selected AA in the rules that describe each specific output class.

| | **A** | **R** | **Q** | **G** | **I** | **L** | **F** | **V** |
|---|---|---|---|---|---|---|---|---|
| **low** | - | - | 2 | - | 2 | 6 | 6 | 1 |
| **medium** | 1 | - | 2 | - | 2 | 2 | 2 | 1 |
| **high** | 1 | 3 | - | 3 | - | - | - | 7 |

Table 5.7. Summarized rules for Class 1, mGluR

| | **A** | **R** | **Q** | **G** | **I** | **L** | **F** | **V** |
|---|---|---|---|---|---|---|---|---|
| **low** | 1 | 1 | 3 | - | - | 4 | 1 | 3 |
| **medium** | 3 | 3 | - | 6 | 4 | 3 | 5 | 4 |
| **high** | 2 | 1 | 1 | 1 | 1 | 1 | 3 | - |

Table 5.8. Summarized rules for Class 2, Calcium Sensing

| | **A** | **R** | **Q** | **G** | **I** | **L** | **F** | **V** |
|---|---|---|---|---|---|---|---|---|
| **low** | - | - | - | 2 | 5 | 1 | 7 | 2 |
| **medium** | 1 | - | - | - | 1 | 5 | - | 7 |
| **high** | 4 | 2 | 6 | 1 | - | 2 | - | 2 |

Table 5.9. Summarized rules for Class 3, $GABA_B$

| | A | R | Q | G | I | L | F | V |
|---|---|---|---|---|---|---|---|---|
| **low** | - | - | - | - | | - | - | - |
| **medium** | - | - | - | 1 | | 1 | - | - |
| **high** | - | - | 3 | | - | 1 | 2 | 3 |

Table 5.10. Summarized rules for Class 4, Vomeronasal.

| | A | R | Q | G | I | L | F | V |
|---|---|---|---|---|---|---|---|---|
| **low** | 3 | 1 | 3 | 2 | - | 2 | - | 4 |
| **medium** | - | 1 | - | 1 | - | - | 2 | - |
| **high** | - | - | - | - | 2 | 1 | 1 | - |

Table 5.11. Summarized rules for Class 5, Pheromone

| | A | R | Q | G | I | L | F | V |
|---|---|---|---|---|---|---|---|---|
| **low** | - | 2 | 3 | 1 | - | - | - | - |
| **medium** | 1 | - | - | 2 | 3 | - | - | 1 |
| **high** | 1 | - | - | 1 | 2 | 3 | 5 | 3 |

Table 5.12. Summarized rules for Class 6, Odorant

| | A | R | Q | G | I | L | F | V |
|---|---|---|---|---|---|---|---|---|
| **low** | 1 | 4 | - | 5 | 5 | 1 | 3 | 1 |
| **medium** | 1 | 2 | 2 | 1 | 2 | 2 | 4 | - |
| **high** | 2 | 2 | 4 | 3 | 1 | 6 | 1 | 9 |

Table 5.13. Summarized rules for Class 7, Taste

### 5.1.2 Rule Extraction for mGluR data set

The complete set of extracted rules is again included in the final Appendix section for the sake of clarity.

The mask extracted for the mGluR data set is the following:

**[0 0 -1 0 -2 0 -3 0 0 -4 -5 0 0 -6 -7 0 0 -8 0 0 1]**

As described in Section 5.1, the negative values in the columns represent the input variables selected by the FIR steps to be the most relevant ones. The zero values correspond to the input variables not selected, and the positive value of 1 describes the output variable.

Then, the selected relevant AAs based on the obtained mask are highlighted in Table 5.14.

| 0 | 0 | -1 | 0 | -2 | 0 | -3 | 0 | 0 | -4 | -5 | 0 | 0 | -6 | -7 | 0 | 0 | -8 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | R | N | D | C | Q | E | G | H | I | L | K | M | F | P | S | T | W | Y | V |

Table 5.14 Correlation of mask values and input variables, regarding the AA alphabet.

The rules follow the same format as seen in Chapter 5.1 – Tables 5.6 and following.

Below, in Tables 5.15 – 5.17, we provide a summary of the obtained rules, regarding the number of appearances of each AA in the rules that describe each specific output class.

| | **N** | **C** | **E** | **I** | **L** | **F** | **P** | **W** |
|---|---|---|---|---|---|---|---|---|
| **low** | - | - | - | - | - | 1 | - | 4 |
| **medium** | 1 | - | - | 1 | 2 | - | - | - |
| **high** | - | - | - | - | - | - | 3 | - |

Table 5.15. Summarized rules for *Group I*

| | N | C | E | I | L | F | P | W |
|---|---|---|---|---|---|---|---|---|
| **low** | 1 | - | 2 | 2 | - | - | 2 | - |
| **medium** | - | - | - | - | - | - | 2 | 1 |
| **high** | - | 2 | - | - | 3 | 3 | - | 3 |

Table 5.16. Summarized rules for *Group II*

| | N | C | E | I | L | F | P | W |
|---|---|---|---|---|---|---|---|---|
| **low** | - | - | - | - | 4 | - | 3 | 1 |
| **medium** | - | - | 1 | 1 | - | 4 | 2 | 2 |
| **high** | | - | 2 | - | - | - | 1 | 1 |

Table 5.17. Summarized rules for *Group III*

## 5.2 Results: Experimental Classification Approach

### 5.2.1 GPCR Class C data set

After obtaining the rules using LR-FIR, we decided to go a step further, and use the FIR methodology to examine what results could be obtained when used for classification.

It has been decided to perform this experiment using the Class C GPCR dataset since it has a higher number of data points (1,510 sequences) than the mGluR subtype (351 sequences). FIR requires as much data as possible representing the system under study.

The fixed variables are the output class, which is always seven classes: the seven class C GPCR subtypes described in previous chapters; and the mask complexity, whose maximum level is defined as 9.

Data Separation

In order to proceed with the classification part, we had to pre-process the data, as for the extraction of rules we have used the full dataset.

As an initial approach, we divided the data set in training and test sets, using a proportion of 70%-30%, respectively, to study different discretization algorithms. For each one of the discretization algorithms supported by VisualFIR, as mentioned in chapter 4.2.1.1, different input variable partitions were selected, specifically three to five, as well as different mask complexities.

Most preliminary experiments, though, failed to make any significant prediction with mask complexity values over levels 4 or 5. This is something we expected, because the input space eventually becomes too big for FIR to handle, thus the inputs cannot be found (all) in the behavioral matrix as the dimensionality increases.

The best performing algorithms were always considering a 3-way partition of the input variables, with the best one being K-means, and secondly Fuzzy C-means and Equal Frequency Partition.

Based on these results, the discretization algorithm (K-means), and thus the mask giving the best accuracy, were chosen for further analysis.

This was done following the idea of *k*-fold cross validation, using 90% as training data, and 10% as test data. Briefly, this is translated as dividing the whole dataset in *k*=10 equal sets, using *k-1* sets as training and *1* set as test, making sure that every time the test set is a different one, not used for testing so far. This is done in order to have all the data involved in the test sets.

It is important to note that the division of the sets was made randomly, making sure that each set (*k*), contains the same proportion of values of each output class (10%).

| No. | Classification Algorithm | Partition of input variables | Mask Complexity | Accuracy (%) |
|---|---|---|---|---|
| 1. | K-Means | 3 | 4 | 64.90 |
| 2. | K-Means | 3 | 5 | 58.94 |
| 3. | K-Means | 3 | 4 | 48.66 |
| 4. | K-Means | 3 | 4 | 62.00 |
| 5. | K-Means | 3 | 4 | 60.66 |
| 6. | K-Means | 3 | 5 | 62.25 |
| 7. | K-Means | 3 | 4 | 49.66 |
| 8. | K-Means | 3 | 4 | 48.34 |
| 9. | K-Means | 3 | 5 | 58.27 |
| 10. | K-Means | 3 | 4 | 69.35 |
| | | | | |
| Mean Accuracy | | | | **58.03** |

Table 5.18. Class C GPCR classification accuracy using FIR with K-means algorithm and 10-fold cross-validation.

Taking into account that using masks of high complexity, usually of levels 5 – 9, a prediction cannot be made, as several instances are found to be missing from the behavioral matrix, (as stated above, due to the fact that the input space becomes too big for FIR to handle), we propose the following approach:

1. Perform a prediction of highest mask complexity, $M_c$=9, and keep the predicted values for the output class
2. Perform a prediction using mask of complexity $M_c$-*1*, substituting only the values that have not been predicted from the previous mask, and that are equal to zero.
3. Repeat Steps 1-2, until all the instances in the output (test set) have been predicted.

This should give us a superior prediction, as we take advantage of the fact that the algorithm only predicts, if a prediction for the entire test set sequences can be made.

This can also be described as a cascade of masks.

Using the above approach, we get the following results, as seen in Table 5.19, taking into account the best fuzzification algorithm (as reported in Table 5.18). The classification results in tables 5.18 and 5.19 are displayed as a mean of accuracy of the 10 folds.

| No. | Classification Algorithm | Accuracy |
|---|---|---|
| 1. | K-Means | 54.96 |
| 2. | K-Means | 55.62 |
| 3. | K-Means | 42.00 |
| 4. | K-Means | 56.00 |
| 5. | K-Means | 56.66 |
| 6. | K-Means | 75.49 |
| 7. | K-Means | 53.64 |
| 8. | K-Means | 54.96 |
| 9. | K-Means | 62.91 |
| 10. | K-Means | 66.22 |
| | | |
| Mean Accuracy | | **57.84** |

Table 5.19. Class C GPCR classification accuracy using FIR with K-means algorithm and the cascade of mask approach (using 10-fold cross-validation).

As seen in Table 5.18 and Table 5.19, the classification results, taking into account the accuracy levels, are more or less the same using the cascade of masks approach.

## 5.3 Discussion

### 5.3.1 Rule Extraction for Class C GPCR

The results summarized in Tables 5.7 – 5.13, contain interesting information regarding the discrimination of Class C GPCR subtypes.

It can be clearly seen that classes/subtypes Calcium Sensing (CaS), Odorant and Taste are difficult to be specifically discriminated from the rest of subtypes, evidence of their internal heterogeneity according to the AAC transformation, coupled with the limited heterogeneity between subtypes, as stated in other research studies [1] [44].

Moreover, mGluR and $GABA_B$ seem to differentiate from the rest subtypes, mainly due to the appearance of -only- high values of amino-acid Arginine (R). This is potentially interesting, because this basic residue has been shown to be crucial for the interaction of the mGluRs with their respective G-protein [19].

Furthermore, from Table 5.10 it can be seen that Pheromone receptors are the only subtype not containing any Alanine (A) and Arginine (R) amino-acids in any of the class' rules, indicating their differentiation from the rest classes.

### 5.3.2 Rule Extraction for mGluR

The results summarized in Tables 5.15 – 5.17 for the specific mGluR subtype and its discrimination into three main groups also reveal some interesting information that seems to be backed by recent studies [5] [44].

As we can see from table 5.16, the rules describing *Group II* can be differentiated from those defining the other two mGluR groups due to the fact that they are the only containing *Cysteine* (C), importantly high values. *Group II* can be also differentiated from the other two groups, regarding Isoleucine values. *Group I* and *Group III* contain medium values of Isoleucine, in comparison to low values in *Group II.*

Moreover, the three groups can be differentiated from five more residues whose values vary in the rules defining each of the three groups. These five AA are:

- Glutamic Acid (E),
    - Group I, no occurrences
    - Group II, low values
    - Group III, medium-high values
- Asparagine (A)
    - Group I, medium values
    - Group II, low values
    - Group III, no occurrences
- Leucine (L)
    - Group I, medium values
    - Group II, high values
    - Group III, low values

- Phenylalanine (F)
    - Group I, low values
    - Group II, high values
    - Group III, medium values

### 5.3.3 Classification for Class C GPCR

As reported in section 5.2.1, from the classification results in Table 5.18 and Table 5.19, the cascade of masks approach does not seem to have a significant effect in the resulting accuracy.

It is evident that the classification results obtained are not good enough to compare the performance of FIR with other classification methodologies, such as SVM for which classification accuracies of 88% have been reported [1], but this is somewhat expected since FIR tries to find a unique model that describes all the output classes. Currently, we are working on a new FIR approach called hierarchical FIR that will allow obtaining several FIR models that explain different output classes in a hierarchical way.

Moreover, datasets regarding the Class C GPCR dataset, but using different transformations, such as auto cross covariance and digram, have reported superior results regarding class classification, in the range of 93% [1].

# Chapter 6

# Conclusions and future work

The use of the FIR methodology to discriminate GPCR subtypes has proved to be difficult due to the nature of the proteins themselves. From the available experimental data, only the 1-gram transformation could be used in FIR. That is, the direct frequencies of appearance of the AAs in the sequence. This is due to the fact that alternative representations would imply the use of input spaces that are much bigger keeping the same number of data sets, leading to a curse of dimensionality.

The obtained Linguistic Rules provide an insight of the discrimination of classes, with respect to the AAs involved in the description of the rules determining an output class, regarding both datasets examined: Class C GPCR and their mGluR subtype.

The results obtained should be the basis for potential ulterior examination by GPCR data curators and bioinformatics domain experts.

As we have also reported, K-means using three partitions for each input variable achieves the highest accuracy amongst the other discretization algorithms tested, with an upper bound of accuracy on the classification experiments regarding Class C GPCR, of around 65%. However, other algorithms are able to produce better accuracy results [41] [42], such as SVM.

Since we have only been able to experiment with two datasets, further experimentation could and should be carried out with FIR, reducing if necessary the input variables when more information is known about the subject.

Furthermore, using an experimental methodology, cascade of optimal masks, so that when an input attribute is missing from the behavioral matrix (causing a FIR prediction stop), its lower masks' level occurrence could be recalled to fill that space, so that the algorithm proceeds and a prediction can be made, has yielded in accuracy results of the same scale.

This approach, as well as the hierarchical FIR approach currently under development, can also be used in the future when different transformations of the data, supported by FIR, are available, since recent studies have shown that using more complex transformations better accuracy results can be obtained.

## References

[1]. C. König, R. Cruz-Barbosa, R. Alquézar and A. Vellido, A, SVM-Based Classification of Class C GPCRs from Alignment-Free Physicochemical Transformations of Their Sequences, New Trends in Image Analysis and Processing – ICIAP 2013, Lecture Notes in Computer Science Volume 8158, 2013, pp 336-343.

[2]. F. Horn, J. Weare, M. W. Beujers, S. Horsch, A. Bairoch, W. Chen, O. Edvarsen, F. Campagne abd G. Vriend. Gpcrdb: An information system for g protein-coupled receptors. Nucleic Acids Res, 26:294-297, 1998.

[3]. J.P. Pin, T. Galvez and L. Praczeau. Evolution, structure and activation mechanism of family 3/c g-protein-coupled receptors. Pharmacology & Therapeutics, 98(3):325-354, 2003.

[4]. M. Rask-Andersen, M. Sallman-Almen, H.B. Schioth, Trends in the Exploitation of Novel Drug Targets. Nature Reviews Drug Discovery 10, 579-590 (2011).

[5]. M.I. Cárdenas, A. Vellido, C. König, R. Alquézar and J. Giraldo, Exploratory visualization of misclassified GPCRs from their transformed unaligned sequences using manifold learning techniques, In F. Ortuño, I. Rojas (eds.): Procs. of the 2nd International Work-Conference on Bioinformatics and Biomedical Engineering, 2014, pp.623-630.

[6]. W.K. Kroeze, D.J. Sheffler and B.L. Roth, G-protein-coupled receptors at a glance, Journal of Cell Science, 2003 116:4867-4869; doi:10.1242/jcs.00902.

[7]. R. Leurs, et al. The histamine H3 receptor: from gene cloning to H3 receptor drugs. Nature Reviews Drug Discovery, 4, 107-120 (2005).

[8].7-TM Receptors, Pharmacology, Tocris Bioscience, http://www.tocris.com.

[9]. F.E. Cellier, A. Nebot, F. Mugica and A. de Albornoz (1992). Combined qualitative/quantitative simulation models of continuous-time processes using fuzzy inductive reasoning techniques. International Journal of General Systems, 24, 95-116.

[10]. Royal Swedish Academy of Sciences. The Nobel Prize in Chemistry 2012 R.J. Lefkowitz, B.K. Kobilka, Retrieved 10 October 2012.

[11]. À. Nebot, S. Medina, and F.E. Cellier, The Causality Horizon: Limitations to Predictability of Behavior Using Fuzzy Inductive Reasoning, Proc. ESM'94, European Simulation MultiConference, Barcelona, Spain, pp. 492-496, 1994.

[12]. K.D. Forbus. Qualitative process theory. Artificial Intelligence, 24, 85-168, 1984.

[13]. T.B. Patel, Single Transmembrane Spanning Heterotrimeric G Protein-Coupled Receptors and Their Signaling Cascades, Pharmacological Reviews, 2004, 56(3), 371-385.

[14]. W. Härdle, Z. Hlavka and S. Klinke, “XploRe® - Application Guide”. Springer Science & Business Media, Nov 16, 2000.

[15]. H. Bräuner-Osborne, P. Wellendorph and A.A. Jensen, Structure, Pharmacology and Therapeutic Prospects of Family C G-Protein Coupled Receptors, Current Drug Targets, 2007, 8, 169-184

[16]. Kniazeff J, Prezeau L, Rondard P, Pin JP, Goudet C. Dimers and beyond: The functional puzzles of class C GPCRs. Pharmacology & Therapeutics, 2011; 130: 9–25.

[17]. J.P. Pin, T. Galvez and L. Prezeau. Evolution, structure, and activation mechanism of family 3/C G-protein-coupled receptors. Pharmacology & Therapeutics, 2003; 98: 325–54.

[18]. K.J. Gregory, E.N. Dong, J. Meiler and P.J. Conn. Allosteric modulation of metabotropic glutamate receptors: structural insights and therapeutic potential, Neuropharmacology 2001; 60: 66-81.

[19]. A. Francesconi and R. M. Duvoisin. Role of the second and third intracellular loops of metabotropic glutamate receptors in mediating dual signal transduction activation. Journal of Biological Chemistry, 1998, 273(10): 5615-5624.

[20]. Narendra Tuteja, Signaling through G protein coupled receptors, Plant Signaling & Behaviour, 2009, 4(10): 942–947.

[21]. H.Hamm. The many faces of G-protein signaling. The Journal of Biological Chemistry, 1966; 273: 669–672.

[22]. R. C. Stevens, V. Cherezov and K. Wüthrich. GPCR Network: a large-scale collaboration on GPCR structure and function. Nature Reviews Drug Discovery. 2013, 12(1):25-34. doi: 10.1038/nrd3859

[23]. D.K. Vassilatis, J.G.Hohmann, H. Zeng, F. Li, J.E. Ranchalis, M. T. Mortrud, A. Brown, S.S. Rodriguez, J.R. Weller, A.C. Wright, J.E. Bergmann and G.A. Gaitanaris, (2003). The G protein-coupled receptor repertoires of human and mouse. Proceedings of the National Academy of Sciences, USA 100, 4903-4908.

[24]. E. Hermans. (2003). Biochemical and pharmacological control of the multiplicity of coupling at G-protein-coupled receptors. Pharmacology & Therapeutics, 99,25-44.

[25]. W.K. Kroeze, S.J. Hufeisen, B.A. Popadak, S.M. Renock, S. Steinberg, P. Ernsberger, K. Jayathilake, H.A. Meltzer and B.L. Roth (2003) H1-histamine receptor affinity predicts

short-term weight gain for typical and atypical antipsychotic drugs. Neuropsychopharmacology 28, 519-526.

[26]. P. Rondard, C. Goudet, J. Kniazeff, J.P. Pin and L. Prezeau. The complexity of their activation mechanism opens new possiblities for the modulation of mGlu and $GABA_B$ class C G protein-coupled receptors. Neuropharmacology 2011; 60: 82–92.

[27]. B.T. Layden, V. Durai, and Jr. W.L. Lowe. G-Protein-Coupled Receptors, Pancreatic Islets, and Diabetes. Nature Education, 2010, 3(9):13

[28]. G.M. Cooper, The Cell: A Molecular Approach, 2nd edition, 2000.

[29]. L. von Bertalanffy. (1969). General System Theory. New York: George Braziller, pp. 39-40

[30]. M. Francetič, M. Nagode, and Bojan, Hierarchical Clustering with Concave Data Sets, *Methodological Notebooks*, 2(2), 2005, 173-193.

[31]. A. Escobet, À Nebot and F.E Cellier. Visual-FIR: A new platform for modelling and prediction of dynamical systems, Proceedings of the SCSC'04: Summer Computer Simulation Conference, San José, CA (USA), 229-234.

[32]. G.J. Klir, and M. Valach, (1967). Cybernetic Modelling. Iliffe, London, and D. Van Nostrand, Princeton, N.J. Translated from the Czech original, published by SNTL, Prague, 1965.

[33]. J. C. Dunn (1973): A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, Journal of Cybernetics 3: 32-57.

[34]. J. C. Bezdek (1981): Pattern Recognition with Fuzzy Objective Function Algoritms, Plenum Press, New York.

[35]. M.I. Heywood, Hard c-means Clustering, CSCI 6506 - Genetic Algorithms and Programming July 2012, Dalhousie University.

[36]. J.B. MacQueen (1967). Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1:281-297.

[37]. P.K. Papasaikas, P.G. Bagos, Z.I. Litou and S.J. Hamodrakas. A novel method for GPCR recognition and family classification from sequence alone using signatures derived from profile hidden Markov models, SAR and QSAR in Environmental Research, 2003;14(5-6):413-20.

[38]. K.A. Johnson, P.J. Conn and C.M. Niswender. Glutamate receptors as therapeutic targets for Parkinson's disease. CNS & Neurological Disorders – Drug Targets, 2009; 8: 475–91.

[39]. C.J. Swanson, M. Bures, M.P. Johnson, A.M. Linden, J.A. Monn and D.D. Schoepp. Metabotropic glutamate receptors as novel targets for anxiety and stress disorders. Nature Reviews Drug Discovery 2005; 4: 131–44.

[40]. H. Wu, C. Wang, K.J. Gregory, G.W. Ha, H.P. Cho, Y. Xia, C.M. Niswender, V. Katritch, J. Meiler, V. Cherezov, P.J. Conn, R.C. Stevens. Structure of a Class C GPCR Metabotropic Glutamate Receptor 1 Bound to an Allosteric Modulator, Nature 511, 557–562 (31 July 2014) doi:10.1038/nature13396.

[41]. R. Cruz-Barbosa, A. Vellido, J.Giraldo. Advances in Semi-Supervised Alignment-Free Classification of G-Protein-Coupled-Receptors. Proceedings of the International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO'13), Granada, Spain, pp. 759-766 (2013).

[42]. C. König, R. Cruz-Barbosa, R. Alquézar and A. Vellido. SVM-Based Classification of Class C GPCRs from Alignment-Free Physicochemical Transformations of Their Sequences. 2nd International Workshop on Pattern Recognition in Proteomics, Structural Biology and Bioinformatics (PR PS BB 2013), 17th International Conference on Image Analysis and Processing (ICIAP) In A. Petrosino, L. Maddalena, P. Pala (Eds.): ICIAP 2013 Workshops, LNCS 8158, pp. 336–343, 2013, Springer.

[43]. GPCRDB, Information system for G protein-coupled receptors, http://www.gpcr.org/7tm/

[44]. M.I. Cárdenas, A. Vellido and J. Giraldo, Exploratory visualization of Metabotropic Glutamate Receptor subgroups through manifold learning, 17th International Conference of the Catalan Association of Artificial Intelligence (CCIA), 2014. In press.

[45]. S. Yin, C.M. Niswender, Progress toward advanced understanding of metabotropic glutamate receptors: Structure, signaling and therapeutic indications. Cellular Signalling, 26(10), 2284-2297.

[46]. M. Masu, Y. Tanabe, K. Tsuchida, R. Shigemoto and S. Nakanishi (1991). Sequence and expression of a metabotropic glutamate receptor, Nature 349 (6312): 760–765. doi:10.1038/349760a0

[47]. A. Vellido, J.D. Martín-Guerrero and P.J.G. Lisboa. Making machine learning models interpretable. In Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning ( ESANN 2012), Bruges, Belgium, pp.163-172.

[48]. A. Vellido, E. Biganzoli and P.J.G. Lisboa. Machine learning in cancer research: implications for personalised medicine. In Proceedings of the 16th European Symposiun on Artificial Neural Networks (ESANN 2008), 55-64

[49]. A. Vellido, P.J.G Lisboa. Neural networks and other machine learning methods in cancer research. In Proceedings of IWANN 2007.LNCS Vol. 4507, 964-971]

[50]. R. Cruz-Barbosa, A. Vellido and J. Giraldo. Advances in Semi-Supervised Alignment-Free Classification of G-Protein-Coupled Receptors, In Proceedings of the International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO'13), Granada, Spain, pp.759-766, 2013.

[51]. J. Garnier, J.F. Gibrat, and B. Robson, GOR Method for Predicting Protein Secondary Structure from Amino Acid Sequence, Methods in Enzymology, 266, pp. 541-553, 1996.

[52]. A.A. Salamov and V.V. Solovyev, Prediction of Protein Secondary Structure by Combining Nearest-Neighbor Algorithms and Multiple Sequence Alignments, Journal of Molecular Biology, 247, 11-15, 1995.

[53]. M.N. Nguyen, J.M. Zurada, and J.C. Rajapakse, Toward Better Understanding of Protein Secondary Structure: Extracting Prediction Rules. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 8 (3) 2011.

[54]. C. Zhou, P. Meysman, B. Cule et al. Mining spatially cohesive itemsets in protein molecular structures. In: Proceedings of the 12th InternationalWorkshop on DataMining in Bioinformatics (BIOKDD 2013), Chicago, IL, 2013. New York: ACM.

[55]. J. He, H. Hu, B. Chen, P.C. Tai, R. Harrison, Y. Pan. Rule Extraction from SVM for Protein Structure Prediction, Rule Extraction from Support Vector Machines. Studies in Computational Intelligence, 80, 2008, pp 227-252.

[56]. T. Ono, H. Hishigaki, A. Tanigami and T. Takagi. Automated extraction of information on protein–protein interactions from the biological literature. Bioinformatics, 17(2), 155-161, 2000.

[57]. Kuipers, B.J., 1986. Qualitative simulation. Artificial Intelligence, 29, 289-338.

[58]. G.J. Klir and B. Yuan. Fuzzy sets and fuzzy logic, Upper Saddle River, NJ: Prentice Hall, 1995.

# Appendix (to Chapter 5)

# Experimental Work: Results and Discussion

### 5.3 Rule Extraction – Full Dataset GPCR Class C Dataset

| RULES |
| --- |
| RULES FOR CLASS 1 |
| IF L IN 1 AND N IN 1-2 AND V IN 1-2 THEN CLASS IN 1 |
| IF R IN3 AND G IN 3 AND L IN 1 AND V IN 3 THEN CLASS IN 1 |
| IF Q IN 2 AND I IN 3 AND F IN 1 AND V IN 3 THEN CLASS IN 1 |
| IF Q IN 2 AND I IN 3 AND L IN 1 AND F IN 1 AND V IN 2 THEN CLASS IN 1 |
| IN Q IN 1 AND G IN 3 AND L IN 1 AND F IN 1 AND V IN 2 THEN CLASS IN 1 |
| IF A IN 3 AND R IN 3 AND Q IN I AND G IN 3 ANDI IN 1 AND L IN 2 AND F IN 2 AND V IN 3 THEN CLASS IN 1 |
| IF R IN3 AND I IN 1 AND L IN 1 AND F IN 2 AND V IN 3 THEN CLASS IN 1 |
| IF R IN 3 AND G IN 3 AND I IN3 AND F IN 1 AND V IN 1 THEN CLASS IN 1 |
| IF A IN 2 AND G IN 3 AND I IN 2 AND L IN 1 AND F IN 1 AND V IN 3 THEN CLASS IN 1 |
| IF Q IN 1 AND G IN 3 AND I IN 2 AND F IN 1 AND V IN 3 THEN CLASS IN 1 |

| RULES FOR CLASS 2 |
| --- |
| IF Q IN 3 AND I IN 2 AND L IN 1 AND F IN 3 AND V IN 1 THEN CLASS IN 2 |
| IF G IN 2 AND I IN 2 AND L IN 1 AND F IN 3 AND V IN 1 THEN CLASS IN 2 |
| IF A IN 2 AND R IN 2 AND Q IN 1 AND G IN 1 AND I IN 2 AND F IN 2 THEN CLASS IN 2 |
| IF G IN 2 AND I IN 3 AND L IN 1 AND F IN 2 AND V IN 3 THEN THEN CLASS IN 2 |
| IF A IN 3 AND R IN 3 AND I IN 2 AND F IN 3 AND V IN 1 THEN CLASS IN 2 |
| IF A IN A AND R IN 2 AND Q IN 1 AND G IN 2 AND I IN 2 AND L IN 1 AND F IN 2 THEN CLASS IN 2 |
| IF A IN 2 AND R IN 1 AND Q IN 1 AND G IN 2 AND I IN 2 AND L IN 2 AND F IN 2 AND V IN 3 THEN CLASS IN 2 |
| IN R IN 1 AND G IN 2 AND I IN 2 AND L IN 3 THEN CLASS IN 2 |
| IF A IN 2 AND R IN 2 AND G IN 3 AND L IN 2 ANDF IN 2 AND V IN 3 THEN CLASS IN 2 |
| IF R IN 3 AND Q IN 1 AND G IN 2 AND L IN 2 AND F IN 1 AND V IN 3 THEN CLASS IN 2 |
| RULES FOR CLASS 3 |
| IF G IN 3 AND I IN 1 AND L IN 2 AND F IN 1 AND V IN 2 THEN CLASS IN 3 |
| IF A IN 3 AND Q IN 3 AND L IN 2 AND F IN 1 AND V IN 3 THEN CLASS IN 3 |
| IF Q IN 3 AND G IN 1 AND F IN 1 AND V IN 2 THEN CLASS IN 3 |
| IF G IN 1 AND I IN 1 AND F IN 1 AND V IN 2 THEN CLASS IN 3 |
| IF A IN 3 AND R IN 3 AND Q IN 3 AND I IN 1 AND L IN 3 AND F IN 1 AND V IN 2 THEN CLASS IN 3 |
| IF Q IN 3 AND I IN 2 AND L IN 2 AND F IN 1 AND V IN 3 THEN CLASS IN 3 |
| IF Q IN 3 AND G IN 3 AND L IN 2 AND F IN 1 AND V IN 2 THEN CLASS IN 3 |
| IF A IN 2 AND Q IN 2 AND I IN 1 AND F IN 1 AND V IN 1 THEN CLASS IN 3 |
| IF A IN 3 AND R IN 3 AND Q IN 3 AND I IN 1 AND L IN 1 AND F IN 1 AND V IN 2 THEN CLASS IN 3 |

| |
|---|
| IF QIN 3 AND I IN 1 AND L IN 3 AND F IN 1 AND V IN 1 THEN CLASS IN 3 |
| IF A IN 3 AND L IN 2 AND F IN 1 AND V IN 2 THEN CLASS IN 3 |
| RULES FOR CLASS 4 |
| IF G IN 1-2 AND I IN 2-3 AND L IN 2 AND F IN 3 AND V IN 1-2 THEN CLASS IN 4 |
| IF Q IN 3 AND G IN 2 AND L IN 3 AND F IN 3 AND V IN 3 THEN CLASS IN 4 |
| RULES FOR CLASS 5 |
| IF A IN 1 AND Q IN 1 AND G IN 1 AND I IN 3 AND L IN 3 AND V IN 1 THEN CLASS IN 5 |
| IF A IN 1 AND Q IN 1 AND G IN 2 AND L IN 2 AND F IN 2 AND V IN 1 THEN CLASS IN 5 |
| IF R IN 2 AND Q IN 1 AND G IN 2 AND L IN 2 AND F IN 2 AND V IN 1 THEN CLASS IN 5 |
| IF A IN 1 AND R IN 1 AND G IN 1 AND I IN 3 AND L IN 2 AND F IN 3 AND V IN 1 THEN CLASS IN 5 |
| RULES FOR CLASS 6 |
| IF A IN 2 AND Q IN 1 AND I IN 3 AND L IN 3 AND F IN 3 AND V IN 3 THEN CLASS IN 6 |
| IF R IN 1 AND Q IN 1 AND G IN 2 AND I IN 2 AND F IN 3 AND V IN 3 THEN CLASS IN 6 |
| IF Q IN 2 AND G IN 2 AND I IN 2 AND L IN 3 AND F IN 3 AND V IN 2 THEN CLASS IN 6 |
| IF A IN 3 AND R IN 1 AND Q IN 1 AND G IN 1 AND I IN 3 ANDL IN 3 AND F IN 3 THEN CLASS IN 6 |
| IF R IN 1 AND G IN 3 AND I IN 2 AND F IN 3 AND V IN 3 THEN CLASS IN 6 |
| RULES FOR CLASS 7 |
| IF Q IN 3 AND G IN 1 AND F IN 2 AND V IN 3 THEN CLASS IN 7 |
| IF A ON 3 AND R IN 3 AND Q IN 3 AND G IN 3 AND I IN 1 AND L IN 3 AND F IN 1 AND V IN 3 THEN CLASS IN 7 |
| IF R IN 3 AND G IN 1 AND L IN 3 AND F IN 2 AND V IN 3 THEN CLASS IN 7 |
| IF R IN 1 AND QIN 3 AND G IN 2 AND I IN 1 AND L IN 3 AND F IN 2 AND V IN 3 THEN CLASS IN 7 |

IF A IN 2 AND G IN 1 AND I IN 2 AND L IN 2 AND F IN 3 AND V IN 3 THEN CLASS IN 7

IF A IN 3 AND R IN 2 AND Q IN 3 AND G IN 3 AND I IN 1NA DL IN 3 AND F IN 2 AND V IN 3 THEN CLASS IN 7

IF R IN 1 AND Q IN 2 AND G IN 1 AND I IN 3 AND L IN 1 AND V IN 1 THEN CLASS IN 7

IF R IN 1 AND I IN 1 AND F IN 1 AND V IN 3 THEN CLASS IN 7

IF A IN 1  AND G IN 3 AND L IN 3 AND V IN 3 THEN CLASS IN 7

IF R IN 1 AND Q IN 2 AND G IN 1 AND I IN 2 AND L IN 2 AND V IN 3 THEN CLASS IN 7

IF R IN 2 AND I IN 1 AND L IN 3 AND F IN 1 THEN CLASS IN 7

## 5.4 Rule Extraction – mGluR Dataset

| RULES |
| --- |
| RULES FOR CLASS 1 |
| IF L IN 2 AND F IN 1 AND W IN 1 THEN CLASS IN 1 |
| IF L IN 2 AND P IN 3 AND W IN 1 THEN CLASS IN 1 |
| IF N IN 2 AND P IN 3 AND W IN 1 THEN CLASS IN 1 |
| IF I IN 2 AND P IN 3 AND W IN 1 THEN CLASS IN 1 |
| RULES FOR CLASS 2 |
| IF L IN 3 AND P IN 1 AND W IN 2 THEN CLASS IN 2 |
| IF C IN 3 AND F IN 3 AND P IN 2 AND W IN 3 THEN CLASS IN 2 |
| IF E IN 1 AND F IN 3 AND P IN 2 THEN CLASS IN 2 |
| IF N IN 1 AND C IN 3 AND E IN 1 AND I IN 1 AND L IN 3 AND F IN 3 AND W IN 3 THEN CLASS IN 2 |
| IF I IN 1 AND P IN 1 AND W IN 3 THEN CLASS IN 2 |
| RULES FOR CLASS 3 |
| IF E IN 3 AND F IN 2 AND P IN 1 THEN CLASS IN 3 |
| IF E IN 3 AND P IN 1 AND W IN 2 THEN CLASS IN 3 |
| IF E IN 2 AND L IN 1 AND P IN 2 THEN CLASS IN 3 |
| IF L IN 1 AND F IN 2 AND P IN 2 THEN CLASS IN 3 |
| IF L IN 1 AND F IN 2 AND P IN 1 AND W IN 2 THEN CLASS IN 3 |
| IJ I IN 2 AND P IN 3 AND W IN 3 THEN CLASS IN 3 |
| IF L IN 1 AND F IN 2 AND W IN 1 THEN CLASS IN 3 |